UCRL

MAZE

An Input Generator for DYNA2D, NIKE2D, TOPAZ2D, and CHEMICAL TOPAZ2D - User Manual

Lawrence A. Sanford

Methods Development Group
Mechanical Engineering

John O. Hallquist
Originator

June 1996

1 INTRODUCTION

MAZE is an interactive program that serves as an input and two-dimensional mesh generator for DYNA2D (Whirley and Engelmann, 1992), NIKE2D (Englemann and Hallquist, 1991), TOPAZ2D (Shapiro and Edwards, 1992), and CHEMICAL TOPAZ2D (Nichols and Westerbrook, 1994). MAZE also generates a basic template for ISLAND (Engelmann and Whirley, 1991) input. Analysts use MAZE to construct the geometry of parts and define the characteristics of the overlaying mesh. MAZE also provides for the specification of boundary conditions, slideline definitions, merging of parts to eliminate nodes along common interfaces, moving of boundary nodes to permit graded zoning, mesh smoothing, the definition of load curves, and the establishment of material properties and equations of state. Although MAZE is comprehensive in its role as an engineering code input generator, it is incomplete. The LLNL engineering analysis codes are undergoing continual improvements and enhancements. As such, MAZE should be viewed as serving as a firm foundation from which to begin preparation for subsequent analysis.

MAZE has been used extensively at the Lawrence Livermore National Laboratory. MAZE has been applied to the generation of input data to study the response of two-dimensional solids and structures undergoing finite deformations under a wide variety of large deformation transient dynamic and static problems and heat transfer analyses. It is used by analysts both at LLNL and elsewhere. The code has evolved to meet changing engineering analysis requirements. Algorithms have been optimized. Versions of MAZE are available for several computing platforms, including CRAY/UNICOS, DEC VAX/VMS, and workstations such as SUN, Silicon Graphics, Hewlett Packard, and IBM. The code has been ported to many other machines, and the use of X-Windows graphics allows the SUN version to port easily to other 32-bit UNIX-based machines. The use of a “single-source” development system assures that all new developments appear simultaneously in all supported code versions.

There are many new features and options in this release of MAZE as well as older undocumented features being freshly presented which improve its performance and versatility in a wide range of applications. Major new features in this release include a greater degree of conformance with the most recent versions of LLNL analysis codes. MAZE is more robust in terms of code integrity and error handling. Analysis code control parameters have been updated. The accuracy and completeness of many material models has been improved. Parameter capabilities, similar to the FORTRAN ‘77 PARAMETER statement, have been added to assist programming and input file preparation. J-Integral crack analysis parameters have been incorporated for use in NIKE2D.

External finite element geometry data can be imported and converted into MAZE line and part definitions. A class of region polygon commands allows users to define a polygonal region in which nodes located therein may be subsequently referenced by the region identifier. Nodes may be assigned to slidelines and boundary commands by virtue of their location rather than specific node number. Additional transition element types have been developed. These new developments substantially enhance the accuracy, efficiency, and user convenience of MAZE for a large class of engineering analysis problems.

MAZE is part of a set of codes developed at LLNL. Other analysis codes include the three-dimensional explicit DYNA3D code (Whirley and Engelmann, 1992), the implicit NIKE3D code (Maker, 1995), the explicit DYNA2D code (Whirley and Hallquist, 1991), and the implicit two dimensional code NIKE2D (Engelmann and Hallquist, 1991). TOPAZ2D (Shapiro and Edwards, 1992) and TOPAZ3D (Shapiro, 1985) are finite element codes for nonlinear heat transfer and field problem analysis. PALM2D (Engelmann, Whirley, and Shapiro, 1990) is a recently developed code for fully coupled thermomechanical analysis. CHEMICAL TOPAZ2D (Nichols and Westerbrook, 1994) incorporates modifications to the heat transfer code TOPAZ2D through the addition of chemical reaction kinetics and chemical mixtures. ORION (Hallquist and Levatin, 1985) is an interactive graphics postprocessor for the two-dimensional codes; INGRID (Christon and Dovey, 1992) is an interactive graphics pre-processor for the three-dimensional codes, and TAURUS (Spelce, 1991) and GRIZ (Dovey and Spelce, 1993) are the interactive graphics postprocessors, for the three-dimensional codes.

The use of MAZE by outside firms has been widespread, and this has played an important role in its development. Many code shortcomings have been discovered and remedied as a direct result of dialog with analysts at LLNL and outside users in industry. In addition, many new capabilities have been incorporated as a result of interaction with both analysts at LLNL and collaborators outside LLNL. It is hoped that the MAZE user community will continue to expand and provide feedback to the author at LLNL, and that improvements made by others will be made available for possible incorporation into future versions of MAZE. This active participation provides important information for future development directions of MAZE.

MAZE was originated and developed by John O. Hallquist at LLNL (Hallquist, 1983). It is with great respect for the accomplishments and the contributions of all MAZE developers which are embodied in MAZE that the author continues the expansion of MAZE's capabilities to meet new challenges in support of computational mechanics.

2 ANALYSIS WITH MAZE

The engineering analysis process begins with a physical description of a problem or system to be studied. First, it is necessary to construct the model. This may be done, in cases of two-dimensional analysis, with the MAZE preprocessor and mesh generator. MAZE prepares an input file for the LLNL analysis codes. This input file is an ASCII text file that may be edited or modified, if desired, prior to submission to the analysis code. MAZE maintains an ASCII text file of the commands used to generate the analysis code input file. This file may be edited or modified, if desired, and re-submitted to MAZE for alternative analysis code input file and mesh generation. Next, the engineering analysis is run to prepare the ASCII printout file and a number of binary plot and restart files. Finally, a post-processor reads the binary plot files and create display and hardcopy graphics output of desired quantities. Each of these steps is described in the following sections.

2.1 Pre-processing and Model Generation

The LLNL two-dimensional analysis codes do not contain any significant model generation capability, and rely almost exclusively upon external software for this task. MAZE significantly reduces the amount of manual preparation of the LLNL analysis code input files. Since the input files are in ASCII text format, many users find it convenient to do all model generation on an engineering workstation, and then transfer the input file to a larger computer to run the analysis.

2.2 Starting a New Maze Analysis

The execution line for MAZE varies slightly depending on the computing platform. On CRAY/ UNICOS and UNIX workstation systems, the execution line is:

MAZE i=*inf* **c=***cfl* **o**=*ofl* **m**=*sfl* **g**=*gfl*

where

inf=input file name containing line definitions
cfl=input file name containing MAZE commands for batch execution
ofl=output file name containing MAZE output - analysis code input file
sfl=output file name containing input commands processed by MAZE, i.e., a “save” file
gfl=input file name containing imported geometry

MAZE may be invoked in an interactive mode using system default file names (see below) by entering the execution line:

MAZE

The specification of other file names is optional, except as noted below. Users may switch between interactive and command modes during a MAZE session through the use of the commands **CFILE** and **TTY**. Line segment information may be read from file *inf* by using the **RLN(S)** and **RSEG** commands. The format of file *inf* is described in section: “Reading Line Segment Data.” Finite element geometry data may be read from file *gfl* and converted into line and part definitions within MAZE. The format of file *gfl* is described in section: “Importing Finite Element Geometry Data.”

Default file names for MAZE files are given in the following table. File names must be unique and can have up to eight characters. File names on DEC VAX/VMS systems will have a file extension of .DAT, however, this extension should never be specified in defining a file name. No file name extensions are expected on non-VMS systems. File sizes are dependent upon the computing platform being used.

Identifier	Default file name	Purpose
inf	(none)	input file containing line definitions
cfl	(none)	input file containing MAZE input commands
ofl	mazout	output file containing results of MAZE execution
sfl	mazsav	output file containing record of all MAZE commands; see NOTE *
gfl	(none)	input file containing geometric finite element model data

*NOTE: It is important to rename the default output command file “mazsav” prior to any resubmission as input for subsequent MAZE analyses. This is necessary as MAZE opens file “mazout” for output of the MAZE commands and any input is overwritten and lost.

2.3 Model Display with MAZE

The workstation-based window system provides a means for a continuous display of model information while running MAZE. Geometries are displayed during all phases in the growth of the mesh as it evolves from an arbitrary collection of lines into part definitions and, ultimately, the completed mesh. Mesh elements and nodes may be displayed. This graphics display is provided as an aid in visualizing the often complex object(s) under analysis. Graphics display is under user control and may be selectively enabled and disabled. This action is often taken when repeatedly making minor variations to an established mesh and the analyst need only view the completed result.

2.4 Engineering Analysis

MAZE prepares an input file for the LLNL analysis codes: DYNA2D, NIKE2D, TOPAZ2D, and CHEMICAL TOPAZ2D. The engineering analysis is then executed which generates the ASCII printout file and a number of binary plot and restart files. The procedural steps of running the analysis codes are presented in the user manuals of the respective codes and will not be discussed.

2.5 Post-processing and Results Display

The LLNL analysis codes may write one or two binary plot databases. The state data plot file family is always created and contains information for complete states at regular intervals; 50 to 100 states of data are typical in a state database. The time history data plot file family contains information for selected nodes and elements, but at every solution state; 1000 to 10,000 states of data are typical in a time history database.

ORION post-processes output from the analysis codes. ORION reads the binary plot databases produced by LLNL codes. ORION allows plotting of color contours, fringes, deformed shapes, and time histories in an interactive graphics environment. ORION can compute a variety of strain measures, momenta, and other response quantities of interest. ORION is supported for the same computing platforms as MAZE: CRAY/UNICOS, DEC VAX/VMS, and SUN, Silicon Graphics, Hewlett Packard, and IBM workstations.

ORION uses the graphics library DIGLIB on all platforms. DIGLIB supports a large number of display and hardcopy graphics devices, including X-Windows and postscript (black and white or color) for hardcopy output.

3 COMMAND DEFINITIONS

3.1 MAZE Phases

MAZE generates meshes that are two-dimensional representations of models composed of ordered quadrilateral elements. Each element is defined by its four corner nodes and its material number. Meshes are created by subdividing the model into regions of interest and then specifying the element distribution within each region. Groups of elements are called “parts”. Parts can be merged together to form the meshed representation of a region of the model. The application of boundary conditions and description of material properties completes the construction of the model. These activities occur within MAZE in three distinct phases: PHASE I allows the definition of the model geometry and the subdivision of regions into parts and elements; PHASE II allows additional geometry refinement and the establishment of interface and boundary conditions, e.g., slideline definitions, part merging to eliminate nodes along common interfaces, movement of boundary nodes for graded zoning, mesh smoothing, and load curve definition. PHASE III allows the assignment and description of material properties to model parts, display, and setting of selected boundary conditions.

Each phase of MAZE activities has its own distinct set of commands. General and Graphical commands apply to all three phases and can be used throughout the MAZE session. The Region Polygon Commands apply to PHASE I and PHASE II. The geometry commands of PHASE I are only applicable in this phase and have no meaning within the subsequent phases. The commands associated with PHASE II and PHASE III follow similar proscriptions. A warning message is displayed if an inappropriate command is encountered within a phase to which it is not assigned. MAZE provides no capabilities to return from PHASE II to PHASE I. This is particularly important to note when constructing a mesh interactively. It is suggested that the user utilize a combination of input files and interactive operations for constructing a model.

MAZE provides specific commands for transitioning to the next phase of operations. The first phase is terminated by the commands **ASSM** or **PASSM** which assemble the parts. Command **WBCD** terminates the second phase. This command instructs MAZE to generate the output file upon receipt of the commands **END** or **T**.

3.2 Format of Commands

Each MAZE command consists of a keyword or symbol. Commands may be truncated to the first four unique characters. The command may also require the specification of one or more data items. All tokens, i.e., commands and data items, must be separated with a minimum of one space. Tokens may not be split across line boundaries. When MAZE is executed interactively, the user will be prompted with the display of a period [.]. Following the acceptance of a command, MAZE will prompt for any required data items by displaying an abbreviated form of the desired data item.

MAZE accepts several formats of numeric data. Floating point data may be expressed in integer format (eg. 2), decimal format (eg. 2.1), or scientific notation (eg. 2.3e-6).

MAZE accepts integer data in the form of floating point data by truncating any fractional component. Numeric data strings should not exceed 25 characters in length. Data items requiring angular input should be specified in units of degrees unless otherwise noted.

All data items required by a MAZE command may be entered immediately following the command. It is not necessary to receive a prompt. Unless otherwise noted, all required data items must be specified. Multiple commands and data may be entered on the same input line. A line of input should not exceed 80 character positions.

MAZE will attempt to identify erroneous commands and data items.

3.3 General Commands

!	Suspend indicator. Each instance of command **!** will cause MAZE to suspend execution for 3 seconds. This command is useful during command file execution in allowing the user to view individual graphics frames before advancing to the next display.
{ ... } C	Comment delimiters. MAZE will not process any input contained between the paired delimiters "{" and "}" or any input following the comment delimiter "C" on the current line of input. A space must preceed comment indicators that are not placed in the first column of the input file. A space must follow the comment delimiter.
CFILE	This command is used to return interactive control of MAZE to the command file specified on the MAZE execute line.
END / T	End / Terminate MAZE.
FLDID / NOFLDID	Include / Omit verbose field descriptors in the MAZE output file.
MAZTL *tolerance*	Establish MAZE tolerance specification. Nonintersecting lines within a tolerance of each other will be treated as intersecting lines by command PART. Nodes closer than *tolerance* will be merged in the part merge commands. Default: 10^{-3}

The MAZE command PARAMETER is used to assign a value to a symbolic name. The feature is particularly valuable in allowing users to use symbolic names throughout the MAZE input file. As such, any modifications to the input file during development or subsequent analysis as might be performed in "what if ..." scenarios may be made in one localized portion of the input deck. This eliminates the need to modify numerous instances of "hard-coded" geometric specifications. Parameters values may be reassigned. The normal placement of the PARAMETER command within a MAZE input file is immediately before the geometry definitions. Command PARAMETER must be placed before the command ASSM.

PARAMETER p_1 $[e_1]$ ... p_n $[e_n]$;

Assign the value of arithmetic expression *e* to parameter *p*. Each parameter *p* must begin with an alphabetic character. Parameters used in subsequent expressions must be preceeded by the percent character "%".

Examples:

```
parameter x [3.14159];
parameter d [180.0 / %x];
parameter y [sin(1.5708 * %c];
parameter v [37 mod 2];
```

Expressions follow mathematical rules of operator precedence:

HIGHEST: ^(Exponentiation), *, /, +, and - :LOWEST

Parenthetic operators (...) may be used to order the expression operations. Expressions may contain operands derived from arithmetic or trigonometric functions. The arithmetic and trigonometric functions (degrees) are:

abs:	Absolute value
exp:	Exponentiation
int:	Converstion to integer
log10:	Common logarithm
log:	Natural logarithm
mod:	Modulo arithmetic
sqrt:	Square root
cos:	Cosine
sin:	Sine
tan:	Tangent
acos:	Arccosine
asin:	Arcsine
atan:	Arctangent

Brackets "[" and "]" are required around each expression. The terminating semi-colon ";" is required.

QUIT

Exit MAZE. No MAZE output is generated.

SHOW p_1 ... p_n;

Display the current values assigned to parameters p_1 through p_n. The terminating semi-colon ";" is required.

TRAP

Terminate MAZE upon occurrence of serious errors.

TTY

Return interactive control to the computer terminal. When MAZE processes the last command in a command file, control is automatically returned to the computer terminal unless the last command is END or T.

TV

Select new graphics output device. MAZE will display a menu of available graphics device drivers.

3.4 Graphics Commands

FRAME / NOFRAME	Enable / disable display of overlaying reference axes and tick marks. Default: FRAME
GRID / NOGRID	Enable / disable display of overlaying grid of orthogonal lines. Default: NOGRID
GSET *r z* δ	Center display at point coordinates (*r,z*) using window $r \pm \delta/2$, $z \pm \delta/2$. Window will remain set until command GSET is re-entered. Default: GSET 0 0 0
Z *r z* δ	Zoom to point coordinates (*r,z*) using window δ. Window size is not retained for subsequent displays.

3.5 Phase I

Phase I mesh generation commonly begins with a series of line definitions that define the geometry of the physical component as well as any necessary construction lines. Generally, points are defined directly in the line definitions rather than with separate point definitions, unless a point is explicitly needed for a special purpose. Using these lines, the element and nodal topologies are then constructed by generating MAZE parts. It is often convenient to establish both line and part definitions with parameters. Parameters allow easy mesh manipulation and/or refinement, especially when constructing complex element topologies or performing parameter studies. There are, of course, many alternative approaches to Phase I mesh generation.

3.5.1 Points

The coordinates of a point may be defined using a symbolic identifier(s) although points are usually defined directly in the line definitions. The symbolic name(s) for a point may be used in any MAZE command requiring the specification of a coordinate pair (*r,z*).

Point Creation

FLPIL l_1 l_2 *r_variable z_variable*	Define a point at the intersection of lines l_1 and l_2. The *r*- and *z*-coordinates of the intersection point will be assigned to the user-defined variables *r_variable* and *z_variable*, respectively. *r_variable* and *z_variable* may be used with command PARAMETER. If lines l_1 and l_2 do not intersect, command FLPIL will not be processed. The reuse of command FLPIL overwrites the current values contained in *r_variable* and *z_variable*.
PTD *symbol r z*	Establish point definition in which *symbol* represents the coordinate pair (*r,z*). *symbol* may be used with command PARAMETER.
PTSV	Display all point definitions established with command PTD.

3.5.2 Lines

A line represents an ordered collection of points. Lines are used by MAZE in the establishment of parts and to define various boundary conditions. MAZE may create lines in several manners. Simple lines, arcs, or straight segments can be generated directly. Existing lines can be copied with or without alteration. Lines can be assembled by concatenating a series of points, segments, and arcs together. In the latter case, command LD is issued to begin the line definition. This is followed by a series of commands that generate points, straight lines, or curved line segments. Additional points can then be added either explicitly or by appending segments to the existing line.

The user may define line numbers in the range: 1 to 589, inclusive. A line may be re-defined, thereby removing its former definition. Lines may not have additional points inserted or deleted once the line has been defined. Most simple meshes should be constructed using smooth lines, i.e., lines with a continuous slope.

Line Graphics

LNON / LNOFF	Enable / disable display of line numbers. Default: LNON
LPON / LPOFF	Enable / disable line plotting commands. Default: LPON
LV	Display all lines.
LVI *n* l_1 ... l_n	Display *n* lines consisting of line numbers l_1 ... l_n.
LVS l_1 l_2	Display all lines between numbers l_1 and l_2, inclusive.
LZOOM l_1	Center the current display on line number l_1.

Line Segment Definitions

LD *n*	Begin definition of line *n*. If line *n* has been previously defined, this command will replace the former line definition. Line *n* may now be constructed by using point and segment commands. $1 \le n \le 589$.
	Straight Lines
LP *n* r_1 z_1 ... r_n z_n	Define *n* points (r_1,z_1) ... (r_n,z_n) to be added to the current line definition.
LPIL l_1 l_2	Define a point for the current line at the intersection of lines l_1 and l_2. The intersection coordinates are displayed.

LRL $n\ r_c\ z_c\ l\ \Theta_1 \ldots \Theta_n$

Define n radial lines of length l originating at point (r_c,z_c) and oriented at angles $\Theta_1^\circ \ldots \Theta_n^\circ$. Positive angles are measured counterclockwise from the positive r-axis. MAZE will assign line numbers to the specified n lines to avoid conflicts with previously defined lines. Command LD is NOT required prior to command LRL.

LVC $\Theta\ l$
LVC $r_1\ z_1\ \Theta\ l$
LVC $r_2\ z_2\ \Theta\ -l$

Define a line segment by a vector of length l oriented at Θ°. The vector begins at the last point defined (first command form) or at (r_1,z_1) (second command form). Θ is measured counterclockwise from the positive r-axis. If $l < 0$, the order of the points is reversed when added to the line, i.e., the computed point is added first, then (r_2,z_2). The first command form adds one point to the line; the second and third forms add two points to the line.

Curved Lines

CLAP $r_1\ z_1\ r_c\ z_c$

Define a circular arc centered at (r_c,z_c) beginning at the last point defined and ending at (r_1,z_1). The arc constructed will proceed counterclockwise and may exceed 180°. Since there is no guarantee that such a circular arc exists, the actual arc that is constructed begins at the last point defined, is centered at (r_c,z_c), and ends on the ray traced from (r_c,z_c) through (r_1,z_1).

CUBIC $\Theta_1\ r_2\ z_2\ \Theta_2$
PCUBIC

Define a free form line segment using a third-order cubic equation beginning at the most recently defined end point (r_1,z_1) with a slope of Θ_1° and ending at (r_2,z_2) with a slope of Θ_2°. All angles are measured in positive degrees counterclockwise from the r-axis. Command PCUBIC displays parameters and constants of the most recent line segment definition generated by CUBIC.

LAD $r_c\ z_c\ \Theta$

Define a circular arc centered at (r_c,z_c) beginning at the last point defined and sweeping through Θ°. A counterclockwise rotation is represented by $\Theta > 0^\circ$.

LAP $r_1\ z_1\ r_c\ z_c$

Define a circular arc centered at (r_c,z_c) beginning at the last point defined and ending at (r_1,z_1). The arc constructed is $\leq 180^\circ$. If the included angle is exactly 180°, the arc crossing the positive r-axis is selected. Since there is no guarantee such an arc exists, the actual arc begins at the last point defined, is centered at (r_c,z_c), and ends on the ray traced from (r_c,z_c) through (r_1,z_1).

LAR $r\ z\ R$	Define a circular arc of radius $\|R\|$ beginning at the last point defined and ending at (r,z). If $R > 0$, the center of the arc lies to the left as one moves from the last point defined to (r,z). If $R < 0$, the center is to the right. For each R, two arcs exist; the arc $< 180^\circ$ is selected. If the included angle is exactly 180°, the arc crossing the positive r-axis is chosen regardless of the sign of R.
LAT $r_1\ z_1\ r_2\ z_2\ R$	Define a circular arc $\leq 180^\circ$ of radius R tangent to the last line segment defined and tangent to line segment joining (r_1,z_1) to (r_2,z_2). The line segments will be extended or truncated, as required. The last point defined, (r_0,z_0), will be replaced by the tangent point on the last line segment defined. Point (r_1,z_1) will be replaced by the tangency point on the line segment $(r_1,z_1 \rightarrow r_2,z_2)$. A cusp will be formed if point (r_2,z_2) is not beyond the tangency point.
LCC $n\ r_c\ z_c\ \Theta_1\ \Theta_2\ r_1\ ...\ r_n$	Define n lines consisting of circular arcs centered at (r_c,z_c) sweeping from angle $\Theta_1{}^\circ$ to $\Theta_2{}^\circ$. The radii of the n lines are represented by $r_1\ ...\ r_n$. MAZE will assign line numbers to the specified n lines to avoid conflicts with previously defined lines. Command LD is NOT required prior to command LCC.
LEP $a\ b\ r_c\ z_c\ \Theta_1\ \Theta_2\ \Phi$	Define an elliptic arc centered at (r_c,z_c) with semi-major axis a and semi-minor axis b. The arc sweeps from $\Theta_1{}^\circ$ to $\Theta_2{}^\circ$, in which both angles are measured from the semi-major axis. Φ° represents the inclination of the semi-major axis measured from the positive r-axis. Positive angles represent counterclockwise rotations. A circular arc will be constructed if $a = b$.
LPT $r_1\ z_1\ r_2\ z_2\ R$	Define a circular arc $\leq 180^\circ$ of radius R beginning at the last point defined and tangent to a line segment joining (r_1,z_1) to (r_2,z_2). The line segment is extended or truncated to begin at the tangency point, i.e., line segment $(r_1,z_1 \rightarrow r_2,z_2)$ is replaced by $(r_t,z_t \rightarrow r_2,z_2)$. If the included angle is exactly 180°, the arc crossing the positive r-axis is selected.
LPTA $r_c\ z_c\ R$	Define a line segment beginning at the last point defined and terminating at its tangency point on an arc of radius R, centered at (r_c,z_c). Two tangent points can exist. The tangent point selected is the first one encountered as the arc sweeps in a counterclockwise direction ($R > 0^\circ$) or in a clockwise direction ($R < 0^\circ$) from the positive r- axis. Only the point of tangency is added to the line definition.

LTAS r_{c1} z_{c1} *rot* r_{c2} z_{c2} R_2

Define a line segment consisting of a circular arc centered at (r_{c1},z_{c1}) followed by a straight line segment. The arc begins at the last point defined and sweeps to the beginning of the straight segment tangent to this arc. Rotation is counterclockwise if *rot* = +1; clockwise if *rot* = -1. The straight segment terminates at its tangency point on a second arc of radius R_2 centered at (r_{c2},z_{c2}). If $R_2 > 0$, the terminating tangent point is the first encountered by a counterclockwise rotation from the positive *r*-axis; if $R_2 < 0$, a clockwise rotation is used.

LTP *r z R*

Define a circular arc $\leq 180^\circ$ of radius *R* tangent to the last line segment defined and terminating at (*r*,*z*). The last line segment is extended or truncated to be redefined as the tangency point. If the included angle is exactly 180°, then the arc which crosses the positive *r*-axis is selected.

ML l_1 l_2

Append line l_2 to l_1. Line l_2 is removed. Duplicated points and overlapping line segments are NOT eliminated. Use command CKL l_1 l_1 after command ML to ensure that the resulting line segment is “smooth”.

Line Segment Definitions: Copied / Offset Lines

LO *l* r_1 z_1 r_2 z_2

Define a line segment by offsetting a segment of line *l* such that the new segment begins at (r_1,z_1) and ends at (r_2,z_2). Points of the source segment that lie between the point closest to (r_1,z_1) and the point closest to (r_2,z_2) are offset in a direction locally normal to the curve in order to create the points of the new segment. The offset distance is calculated by bilinear interpolation between the distance from the curve to (r_1,z_1) and the distance to (r_2,z_2). Command LO does not necessarily make straight lines into other straight lines. New endpoints (r_1,z_1) and (r_2,z_2) should be oriented in the “same direction” as the endpoints of the original line segment.

LOD *l* δ

Define a line segment by offsetting line segment *l* a distance δ. This offset is applied in a direction locally normal to the given curve. Positive δ offsets *l* to the left as one moves along the segment in the direction in which it was originally defined. Negative δ offsets *l* to the right. If the initial or final point(s) of the original curve lie on the vertical axis, but the corresponding initial or final segment does not lie entirely on this axis, the initial or final offset is vertical instead of normal to the curve. This change affects the entire new line segment.

LSTL $l\ \Delta r\ \Delta z$ — Define a line segment by translating the entire line l an offset of Δr and Δz.

LT $l\ \Delta r\ \Delta z$ — Translate line l by offset Δr and Δz.

LTM $n\ l_1 \ldots l_n\ \Delta r\ \Delta z$ — Translate n lines $l_1 \ldots l_n$ by offset Δr and Δz.

LTS $l_a\ l_b\ \Delta r\ \Delta z$ — Translate consecutive lines $l_a\ l_b$ by offset Δr and Δz.

VLOD $l\ \delta_1\ \delta_2$ — Define a line segment by offsetting a distance δ_1 from the first point and a distance δ_2 from the last point of line l. Intermediate points are linearly interpolated between δ_1 and δ_2. This offset is applied in a direction locally normal to the given curve. Positive values of δ offset the line segment to the left as one moves along the segment in the direction in which it was originally defined. Negative values of δ offset the line segment to the right.

Line Segment Definitions: Tab Cell Data

LTBC $n\ \Theta\ \Delta\Theta\ S\ R_1 \ldots R_n$ — Define a line segment with tab cell data. Tab cell data consists of n radii, each separated by $\Delta\Theta°$, starting at $\Theta°$. Each radius is scaled by S. Positive angles represent counterclockwise rotations. The first point created will be at $\Theta°$. A maximum of 360 points may be defined. The scale factor is incorporated into the stored radii values. This may be used in subsequent LTBO commands.

LTBO $m_1\ \delta_1 \ldots m_k\ \delta_k$ — Define a line segment by offsetting the last line segment defined with the commands LTBC or LTBO. δ_1 is added to the radii of the first m_1 points; δ_k is added to the radii of the next m_k points. NOTE: $n = m_1 + \ldots + m_k$, where n is defined by the most recent LTBC command. Only the segment generated with the last LTBC/LTBO command is offset, not the entire line containing the segment.

.

Auxiliary Line Operations

CKL $l_1\ l_2$ — Examine all lines from l_1 to l_2 (inclusive) to eliminate all external angles ≥ 120° and coalesce all duplicated points.

DELETE l — Delete line l.

LPRI l — Print the coordinates of line l on the terminal.

MLN — Print the maximum line number used.

NDL — Print the numbers of all lines that have been deleted.

3.5.3 Parts

A part is an ordered set of elements having a unique material number and part identification number. Parts are defined by specifying their boundaries as an ordered set of lines (or points), their material number, and specifiers denoting the pattern for subdividing the part into elements. Parts are numbered consecutively and automatically by MAZE.

MAZE creates elements within a part: boundary nodes are established by subdividing the edges of the part, using arc length or angular position; interior nodes are computed by interpolation of the edge nodes. Elements are either triangles or quadrilaterals. Elements may be subdivided for the creation of transition regions.

Users may select parameters governing the subdivision of edges, weighting the element sizes with respect to one end or one corner of the part, specifying explicit points (via line definitions) for use in subdividing the part edges, and requesting that the number of elements change from one edge to the opposing edge on the other side of the part.

The lines comprising a part must be specified in counterclockwise order. The lines must intersect each other at the corners of the part but need not end at the corners.

Part Graphics

LVPV	Display all lines and parts. Node points will be shown within parts.
PNON / PNOFF	Enable / disable display of part numbers within plots. Default: PNON
PPON / PPOFF	Enable / disable display of part plotting. Default: PPON
PV	Display all parts. Elements will be shown within parts.
PVI $n\ p_1 \ldots p_n$	Display n parts consisting of part numbers $p_1 \ldots p_n$. Elements will be shown within parts.

Evenly Weighted Zoning - Quadrilateral Regions

PART L_1 L_2 L_3 L_4 *material k m*

Define the four sided region of *material* bounded by lines L_1, L_2, L_3, and L_4 to be a part with *k* elements along sides L_1 and L_3, *m* elements along sides L_2 and L_4. If *k* or *m* is equal to zero, then the points L_1, L_3 or L_2, L_4 will be used as nodes. The number of elements will then be one less than the number of points in these lines.

QUAD r_1 z_1 ... r_4 z_4 *material k m*

Define the four sided region of *material* bounded by corners r_1,z_1 ... r_4,z_4 to be a part with *k* elements along sides r_1,$z_1 \rightarrow r_2$,z_2 and r_4,$z_4 \rightarrow r_3$,z_3, *m* elements along sides r_1,$z_1 \rightarrow r_4$,z_4 and r_2,$z_2 \rightarrow r_3$,z_3.

RECT r_1 z_1 r_3 z_3 *material k m*

Define the rectangular region of *material* bounded by opposite corners r_1,z_1 and r_3,z_3 to be a part with *k* elements in the *r* direction, *m* elements in the *z* direction. The sides of the part will be parallel to the *r*- and *z*-axes.

Quadrilateral Transition Parts

The following transition commands are entered as a preface to a PART, QUAD, or RECT command. These commands do NOT apply to triangular parts or parts with variable zoning.

T12

Causes the row of elements along part side L_3 to be subdivided into two times the number of elements in the rows "parallel" to sides L_1 and L_3. If L_3 contains an odd number of elements, then one additional element is created on side L_2.

T13

Causes the row of elements along part side L_3 to be subdivided into three times the number of elements in the rows "parallel" to sides L_1 and L_3.

T21

Causes the row of elements along side L_3 to be subdivided into one-half the number of elements in the rows "parallel" to sides L_1 and L_3. Side L_1 must contain a quantity of elements which is a multiple of two.

T31

Causes the row of elements along side L_3 to be subdivided into one-third the number of elements in the rows "parallel" to sides L_1 and L_3. Side L_1 must contain a quantity of elements which is a multiple of three.

TRANS

Changes the part to one containing $k + m$ elements along sides L_1 and L_2 and *m* elements along sides L_3 and L_4. Zoning subdivides the specified region into three quadrilateral subregions.

Evenly Weighted Zoning - Triangular Regions

PART L_1 L_2 L_3 L_3 *mat k m*

Define the three sided region of material *mat* bounded by lines L_1, L_2, and L_3 to be a part with $k + m$ elements along sides L_1 and L_2, $2m$ elements along side L_3. Zoning is accomplished by subdividing the specified region into three quadrilateral subregions.

PART L_1 L_2 L_3 *0 mat k m*

Define the three sided region of material *mat* bounded by lines L_1, L_2, and L_3 to be a part with k elements along side L_1, m elements along sides L_2 and L_3. k triangular elements will exist at the intersection of lines L_2 and L_3. The vertex with triangular elements will only be identified as a corner node if there are one or three elements at the vertex.

TRIQ r_1 z_1 r_2 z_2 r_3 z_3 *mat k m*

Define the three sided region of material *mat* bounded by (r_1,z_1), (r_2,z_2), and (r_3,z_3) to be a part with $k + m$ elements along side $(r_1,z_1 \rightarrow r_2,z_2)$ and side $(r_2,z_2 \rightarrow r_3,z_3)$, $2m$ elements along side $(r_3,z_3 \rightarrow r_1,z_1)$. Points must be specified in counterclockwise order.

TRIT r_1 z_1 r_2 z_2 r_3 z_3 *mat k m*

Define the three sided region of material *mat* bounded by (r_1,z_1), (r_2,z_2), and (r_3,z_3) to be a part with k elements along side one $(r_1,z_1 \rightarrow r_2,z_2)$, m elements along side two $(r_2,z_2 \rightarrow r_3,z_3)$ and side three $(r_3,z_3 \rightarrow r_1,z_1)$. k triangular elements are created at vertex (r_3,z_3). The vertex with triangular elements will only be identified as a corner node if there are one or three elements at the vertex.

Regions Bounded by a Line and an Arc or One Arc

PART L_1 L_2 L_2 L_2 *mat k m*

Define the region of material *mat* bounded by lines L_1 and L_2 (one line must be straight and one line must be an arc) to be a part with $m + k + k + m$ elements ($k \geq 3$; $m \geq 2$) along sides L_1 and L_2. Zoning is accomplished by subdividing the specified region into six quadrilateral subregions containing a total of $2m^2 + 4km$ elements.

PART L_1 L_1 L_1 L_1 *mat k m*

Define the region of material *mat* bounded by elliptic arc L_1 to be a part with $2(m + k + k + m)$ elements ($k \geq 3$; $m \geq 3$) along arc L_1. Zoning is accomplished by subdividing the specified region into twelve quadrilateral subregions containing a total of $4m^2 + 8km$ elements.

Nodal Spacing Weighted in One or Both Directions

PART ... } {*material -k m* R_1
QUAD ... } {*material k -m* R_2
RECT ... } {*material -k -m* R_1 R_2

Note: PART, QUAD, RECT, and their arguments are as defined for quadrilateral regions.

Define a four sided part of *material* with nodal spacing and element sizing that transitions smoothly across the part. R_1 is the ratio of the first segment length, i.e., node spacing, to the last segment length along edges L_1 and L_3. R_2 is the ratio of the first segment length to the last segment length along edges L_2 and L_4. The "first" direction is from side L_1 to side L_3; the "last" direction is from side L_4 to L_2.

Nodal Spacing Explicitly Weighted Along Each Side

PART ... }
QUAD ... } *-material k m* R_1 ... R_4
RECT ... }

Note: PART, QUAD, RECT, and their arguments are as defined for quadrilateral regions.

Define a four sided part of *material* with independent nodal spacing and element sizing that transitions smoothly across the part. R_1 ... R_4 are the ratios of the first segment length, i.e., node spacing, to the last segment length from: corner 1 $\rightarrow$ corner 2; corner 2 $\rightarrow$ corner 3; corner 3 $\rightarrow$ corner 4; and corner 4 $\rightarrow$ corner 1.

Line Points to Define Nodal Spacing Locations

PART L_1 L_2 L_3 L_4 {*material 0 m*
{*material k 0*
{*material 0 0*

Note: PART, QUAD, RECT, and their arguments are as defined for quadrilateral regions.

Define a four sided part of *material*. If $k = 0$, each point of lines L_1 and L_3 will become a node. If there are j points along L_1 and L_3, there will be $j - 1$ elements along these edges. m elements will be created along edges L_2 and L_4. Conversely, if $m = 0$, each point of lines L_2 and L_4 will become a node, and k elements will be created along edges L_1 and L_3. For $k = 0$ and $m = 0$, all points along each edge will become nodes. The total number of points along corresponding edges must be the same. Corner points will always be nodes and only those points located between corner points will become nodes. No line may extend more than one point beyond a corner.

PART -L_a L_b L_c L_d *mat k m* $n^a{}_1$ … $n^a{}_{pa\text{-}2}$

PART -L_a L_b -L_c L_d *mat k m* $n^a{}_1$… $n^a{}_{pa\text{-}2}$ $n^c{}_1$… $n^c{}_{pc\text{-}2}$

PART L_a -L_b L_c L_d *mat k m* $n^b{}_1$ … $n^b{}_{pb\text{-}2}$

…

et cetera

Define a four sided part of material *mat*. Along side L_a place $n^a{}_1$ elements between the first two points, $n^a{}_{pa\text{-}2}$ elements between the next *pa-2* points, et cetera. The number of elements in the last interval is unspecified in order to satisfy *k*. Nodal distribution along the remaining sides is unaffected. Not all subdivisions need be defined: specifying $n^a{}_1$ $n^a{}_2$ *0* will define the number of elements between points 1 and 2, then 2 and 3 of side L_a. The remaining *k* - $n^a{}_1$ - $n^a{}_2$ elements will be equally spaced between the third point and corner 2. No line may extend more than one point beyond a corner.

PART -L_a L_b L_c L_d *mat* -*k m* R_1 $n^a{}_1$ … $n^a{}_{pa\text{-}2}$

Define a four sided part of material *mat* using previously described element and nodal spacing procedures. Weighted nodal spacing will apply only to side L_c since the explicit nodal spacing takes precedence over side L_a.

Nodal Spacing Based on Angular Position

AZOFF

Disable equal angular zoning.

AZON *n* S_1 … S_n r_c z_c

In subsequent PART commands, nodes will be distributed along sides $S_{i;\,(1 \le i \le 4)}$ using equal angular spacing based on the center point (r_c, z_c). This command will remain in effect until either another AZON or AZOFF command is given. This command will override explicit nodal spacing specifications.

Part Duplications

CLONE *n mat* Δr Δz Θ

Define a part of material *mat* by duplicating part *n*. The duplicated part will be translated by (Δr,Δz) and rotated Θ° counterclockwise from the positive *r*-axis. The axis of rotation is perpendicular to the *r*-*z* plane and passes through (0,0).

RFLIP *n mat*

Define a part of material *mat* by duplicating part *n*. The duplicated part will be rotated about the *r*-axis.

ZFLIP *n mat*

Define a part of material *mat* by duplicating part *n*. The duplicated part will be rotated about the *z*-axis.

Auxiliary Part Commands

AOR Θ	MAZE will attempt to establish a node at the vertex of "sharp" angles $< \Theta^\circ$ in part boundary lines in order that the angles be preserved within the part description. Angles $\geq \Theta^\circ$ may be smoothed in the part description. Angle preservation may be eliminated by setting $\Theta = 0^\circ$. Default: $\Theta = 120^\circ$
BPN *n*	Number parts consecutively beginning with part number *n*. This command must be invoked prior to defining any parts. Default: $n = 1$
DP *m*	Delete part *m*. Number *m* will not be re-used if subsequent parts are defined.
FIXP *n*	Set *r* and *z* constraints for part *n*.
GEOZ	Switch between algebraic and geometric zoning. Default: algebraic zoning
MG *n m*	Merge interface nodes of parts *n* and *m* having the same coordinates. Parts *n* and *m* must have the same material number. The merged part will become part *n*; part *m* will no longer exist. Number *m* will not be re-used if subsequent parts are defined.
NLD *L m* NLD *L -m* R_1 NLD *L 0* NLD *-L m* $n_1 \ldots n_{p-2}$	Establish a node line definition. The node line will consist of $m + 1$ nodes spaced along line *L*. These nodes will be written into the MAZE output file. Operations of $m < 0$, $m = 0$, and $L < 0$ function as described in the PART commands. Node lines do not allow equal angle zoning.
REXT *n rx*	Scale part *n* to extend *rx* units in the *r* direction. This command may translate the part. Command RMIN will return the part to its original location.
RMIN *n rmin*	Translate part *n* to have minimum r-coordinate value *rmin*.
ZEXT *n zx*	Scale part *n* to extend *zx* units in the *z* direction. This command may translate the part. Command ZMIN will return the part to its original location.
ZMIN *n zmin*	Translate part *n* to have minimum z-coordinate value *zmin*.

3.6 Regions

The class of region commands allows users to define a polygonal region surrounding an existing item of MAZE mesh geometry as an alternative method of referencing boundary nodes. Region polygons may be defined for points, line segments, curves, and arcs. All boundary nodes enveloped within the polygon's *radius* or *tolerance* area may then be referenced via the region identification number without regard to the individual node numbers. Users may define slidelines and boundary conditions by virtue of a boundary node's containment within the region polygon rather than specific node number. This capability will retain established slidelines and boundary conditions during on-going analysis wherein node numbers might be altered, e.g., the inclusion of another part within the analysis problem or modification of the mesh density, without requiring modification of the MAZE input file to reflect the updated node numbering sequence.

A single region may envelope coincidental nodes of adjacent parts. Slideline and boundary conditions may still be established using the class of region commands. For example, to establish a slideline between Part 1 and Part 2 sharing a coincident side between coordinates (4,0) and (4,5), the user might enter the following sequence of commands:

```
linr 5 0.05 4.0 0.0 4.0 5.0
...
p 1 b
slvr 5
p 2 b
msrr 5
```

The region commands which define a specific region are both PHASE I and PHASE II commands. However, MAZE commands which reference the polygonal regions and the nodes contained therein, e.g., commands MSRR, BCRN, et cetera, are PHASE II commands.

ARCR *region radius a b* r_c z_c Θ_1 Θ_2 ϕ	Define arc *region* of *radius* defined by semi-major axis *a*, semi-minor axis *b*, and centered around (r_c,z_c). The arc will sweep from $\Theta_1°$ to $\Theta_2°$ and be placed $\phi°$ between the semi-major axis and the *r*-axis.

BCRN *symbol region*	Assign the node number of a single boundary node contained within a previously defined *region* to *symbol*. If *region* contains more than one boundary node command BCRN will not process any data. This command is a variant of command PARAMETER. Any *symbol* used in subsequent expressions must be preceeded by the percent character "%". This command must be preceeded by command B.
CRVR *region tolerance n* r_1 z_1 ... r_n z_n	Define curve *region* of *tolerance* defined by *n* coordinated pairs r_1 z_1 ... r_n z_n
LDR *region tolerance line*	Define line *region* of *tolerance* bounding the *line* previously defined by command LD.
LINR *region tolerance* r_1 z_1 r_2 z_2	Define linear *region* of *tolerance* bounding the line connecting (r_1,z_1) and (r_2,z_2).
PNTR *region radius r z*	Define point *region* of *radius* centered around point (*r,z*).
RV	Display all previously defined regions.
RVI *n region*$_1$ ... *region*$_n$	Display *n* previously defined regions: *region*$_1$ ... *region*$_n$.

3.7 Transition From Phase I To Phase II

Mesh Assembly

ASSM	Assemble mesh from all previously defined parts. Command ASSM will not merge parts.
PASSM $n\ p_1 \ldots p_n$	Assemble mesh from a subset of n parts consisting of part numbers $p_1 \ldots p_n$. Command PASSM will not merge parts.

3.8 Phase II

Commands that act upon boundary nodes must be assigned in one of three manners: a. node-to-node; b. part side; or c. region. The boundary node commands will be applied: a. from node *m* to node *n*; b. to nodes along part side *s*; or c. to nodes within region *r*. These commands must be preceeded by the command sequence:

P *part_number* B

Selected section headings and MAZE command syntax definitions may specify one or more analysis codes. Each code is identified to indicate the restricted command(s) applicability. TOPAZ2D and CHEMICAL TOPAZ2D codes will collectively be called TOPAZ.

3.8.1 General Commands

B	Establish and display boundary nodes defining sides of part *n*. Command B must follow command P and must preceed all commands associated with boundaries.
BLEND *option*	Set smoothing option for use by commands GS and S: EQ. 0.0: equipotential smoothing EQ. 1.0: isoparametric smoothing $0.0 < option < 1.0$: combined blending This command must be used before commands GS and S.
CBNR *region r z*	Change coordinates of single boundary node in *region* to coordinates (*r,z*).
CN *m r z*	Assign node *m* new coordinates (*r,z*).
CNMN *m n*	Set node *m* coordinates to be same as node *n* coordinates.
FLCD *id* t_0 t_n *n [function(*t*)]*	Define load curve *id* containing time-function points. *n* function points are determined by evaluating *function(***t***)* within the range t_0 through t_n. Only parameter "t" is valid for evaluation of the time function. The function expression must be enclosed within brackets "[" and "]".
FLIP	Interchange axes of symmetry.
GS	Smooth all parts.
LCD *id m* t_1 f_1 ... t_m f_m	Define load curve definition *id* containing *m* pairs of time(*t*)-function(*f*) points.
P *n*	Set part *n* for modification.
R	Restore mesh to the condition existing after executing commands ASSM or PASSM.

S	Smooth mesh of part *n*. This command must be preceeded by command P.
SIDE	Establish boundary defining sides of part *n*. Display boundary with side identification only. Command SIDE must follow command P and must preceed all commands associated with boundaries.

3.8.2 Graphics Commands

A	Display all slidelines.
AML	Display all master sides of slidelines.
AS *m n*	Display slidelines *m* through *n*.
ASL	Display all slave sides of slidelines.
CNPO / CNPS	Display / do NOT display corner nodes. These commands must be proceeded by command B.
DBN	Delete boundary nodes from boundary plots generated by command B. Re-entering command will restore boundary nodes in subsequent displays.
DSN	Delete side numbers from boundary plots generated by command B. Re-entering command will restore side numbers in subsequent displays.
ELPLT	Display element numbers on mesh of materials.
ELPM $n\ p_1 \ldots p_n$	Plot element numbers on mesh of *n* parts $p_1 \ldots p_n$.
G	Display complete mesh grid with part numbers.
NDPLT	Display node numbers on mesh of materials.
NDPM $n\ p_1 \ldots p_n$	Plot node numbers on mesh of *n* parts $p_1 \ldots p_n$.
LCV	Display all load curves.
LCVI $n\ lc_1 \ldots lc_n$	Display *n* load curves consisting of curves $lc_1 \ldots lc_n$.
O	Display complete outline of parts with part numbers.
OG	Display complete outline of parts with part numbers surperimposed over grid.
TE $r\ z\ \Delta l$	Display element numbers and coordinates of all element centroids contained within window ($r \pm \Delta l/2$, $z \pm \Delta l/2$).

TN r z Δl	Display node numbers and coordinates of all nodes contained within the window ($r \pm \Delta l/2$, $z \pm \Delta l/2$).
TNC n	Display nodal coordinates of node n. This command must be preceeded by command P.

3.8.3 Merging

All nodes closer than the tolerance specified by command MAZTL will be merged in the part merge commands.

GM p_n p_m	Merge all common interface nodes of parts p_n and p_m. Coordinates of edge nodes on the part containing the least number of nodes will remain unchanged. Only the beginning and ending nodes of the interface are required to have common coordinates. A minimal number of triangular elements may be created in order to successfully merge all nodes along the common interface. If these parts have the same material number, p_m will be merged into p_n. Upon completion of the merge only the combined p_n will remain. The identify of p_m will be lost.
M p_n p_m	Merge all common interface nodes of parts n and m having same coordinates. Part identity is maintained regardless of the material numbers of each part.
MG p_n p_m	Merge all common interface nodes of parts n and m having same coordinates. If these parts have same material number, part m will be merged into part n. Upon completion of the merge operation only the combined part n will remain. The identify of part m will be lost.
MGM m p_0 p_1 ... p_m	Merge m parts with part p_0 to form a new part. The new part will be identified with part number of p_0. m merge operations will proceed in accordance with the ordering of parts p_1 ... p_m. Upon completion of each successive merge operation, the next part in the list must share all common interface nodes with the newly merged part. All parts must have the same material number.
MGM -p_0 p_m	An alternative form of command MGM in which parts p_0 to p_m represent consecutive parts of the same material.
MGN n m	Merge nodes n and m. The coordinates of the remaining node will be assigned the coordinates of node n. Nodes n and m do not need to be close.

3.8.4 Nodal Modification and Spacing

BD *m n* BDS *s*	Remove kinks from boundary. Kink removal is performed in a counterclockwise direction.
EA *m n* EAS *s*	Assign spacing of boundary nodes such that the included angle at point (0,0) between any two adjacent boundary nodes is equal.
ER *m n* ERS *s*	Assign equal spacing of boundary nodes in *r* direction in a counterclockwise direction.
ES *m n* ESS *s*	Assign equal spacing of boundary nodes in a counterclockwise direction.
EZ *m n* EZS *s*	Assign equal spacing of boundary nodes in *z* direction in a counterclockwise direction.
VA *m n ratio* VAS *s ratio*	Assign spacing of boundary nodes in the *ratio* of the first angle at point (0,0) to the last segment angle.
VS *m n ratio* VSS *s ratio*	Assign variable spacing of boundary nodes. Spacing is assigned in the *ratio* of first segment length to last segment length.

3.8.5 Nodal Boundary Conditions: DYNA2D - NIKE2D

NBC *m n code* NBCR *r code* NBCS *s code*	Define boundary nodes to be assigned boundary condition *code*: EQ. 0: no constraint EQ. 1: *r* direction constraint EQ. 2: *z* direction constraint EQ. 3: both *r* and *z* direction constraints If the *code* is not one of the above, the value specified will represent the angle (degrees) between the positive horizontal axis and the direction of motion along a sliding boundary. If an angle of 0°, 1°, 2°, or 3° is desired, the user must specify an approximation to the desired angle, e.g., specify 1.9999... or 2.00...1, for an angle of 2°.
NBCC *corner code*	Define boundary constraint on *corner* to be assigned boundary condition *code*. See command NBC for code descriptions.
RCON *R*	Constrain in the horizontal direction all nodes located on the line $r = R$.
ZCON *Z*	Constrain in the vertical direction all nodes located on the line $z = Z$.

3.8.6 Nodal Loads: DYNA2D - NIKE2D

CNL *m n k* r_1 r_2 *i*
CNLC *corner_node k* r_1 r_2 *i*
CNLS *s k* r_1 r_2 *i*

Assign concentrated nodal loads to act in direction *i*:
EQ. 1: *r* direction
EQ. 2: *z* direction
Nodal loads are assigned in a counterclockwise direction. Nodal loads vary in time according to load curve *k*. The scale factor on load curve *k* varies linearly from r_1 to r_2.

PBC *m n k* r_1 r_2
PBCR *r k* r_1 r_2
PBCS *s k* r_1 r_2

Assign pressure loads boundary condition. Pressure loads are assigned in a counterclockwise direction. Pressure loads vary in time according to load curve *k*. The scale factor on load curve *k* varies linearly from r_1 at boundary node *m* to r_2 at boundary node *n*. If load curve $k < 0$, then $k = |k|$ and the loading flag switches the pressure load to a shear load.

SBC *m n k sf pf* r_c z_c Θ *radius s*
SBCR *r k sf pf* r_c z_c Θ *radius s*
SBCS *side k sf pf* r_c z_c Θ *radius s*

Apply a spatially nonlinear pressure boundary condition on a part. The pressure magnitude varies temporally according to load curve *k*. The spatial pressure variation is determined by the geometric scale function *sf* and is dependent on the distance *d* of the boundary nodes from a line which passes through (r_c,z_c) at angle Θ°. A projection flag, pf (0: off; 1: on), will multiply the scale function by the dot product of the line normal and the local boundary normal. The *radius* denotes a cutoff distance limiting the non-zero scale function range (sf = 0 for $d >$ *radius*). If load curve $k < 0$, then $k = |k|$ and the loading flag switches the pressure load to a shear load.

scale function *sf*:
EQ. 1: 1
EQ. 2: cos (*d/s*)
EQ. 3: $\cos^2$ (*d/s*)
EQ. 4: 1 / (exp((d^2/s^2) / 2)) Gaussian
EQ. 5: $1/d^s$
EQ. 6: cos ($\sin^{-1}$(d/s))

3.8.7 Prescribed Nodal Kinematics

DBC m n k r_1 r_2 i DBCR r k r_1 r_2 i DBCS s k r_1 r_2 i (NIKE2D)	Assign displacement time history to boundary nodes to act in direction i: EQ. 1: r direction EQ. 2: z direction Displacement time histories are assigned in a counterclockwise direction. Displacement time histories loads are specified by load curve k. The scale factor on k varies linearly from r_1 to r_2.
IAV ω r_c z_c	Set initial angular velocity, ω, about the normal axis passing through (r_c,z_c) of all parts. This command is applicable to plane strain and plane stress geometries only.
IV v_r v_z	Set initial velocity components of all parts to (v_r,v_z).
IVN m n v_{rm} v_{zm} v_{rn} v_{zn}	Set initial velocity components of nodes m through n to (v_{rm},v_{zm}) and (v_{rn},v_{zn}), respectively. Initial velocities of nodes between m and n are linearly interpolated.
IVP n v_r v_z	Set initial velocity components of part n to (v_r,v_z).
NRBN m n NRBR r NRBS s (DYNA2D)	Assign a non-reflecting boundary condition. Boundary condition is assigned to nodes in a counterclockwise direction.
VBC m n k r_1 r_2 i VBCR r k r_1 r_2 i VBCS s k r_1 r_2 i (DYNA2D)	Assign velocity time history to act in direction i: EQ. 1: r direction EQ. 2: z direction Velocity time histories are assigned in a counterclockwise direction. Velocity time histories are specified by load curve k. The scale factor on k varies linearly from r_1 to r_2.

3.8.8 Slideline Definitions

Slideline definitions are established through a series of steps. The user must first define the slideline number and type of slideline (command SLN). The entire master (or slave) line is then described followed by the entire slave (or master) line using commands MSRS, MSRR, MSRS, commands SLV, SLVR, SLVS, or commands SLBMP, SLBP. The nodes along (or sides of) the master (or slave) side must be defined such that as one travels around the completed slideline, from the first node defined to the last node defined, the direction around the part(s)s is counterclockwise. Furthermore, when commands SLV and MSR are used, MAZE automatically reverses the nodal ordering in the slideline defintion which is output if the direction from node *n* to node *m* is clockwise.

Slideline definitions must be preceeded by command SLN. If a slideline number has previously been established, reissuing the defining command replaces the former definition.

MSR *m n* MSRR *r* MSRS *s*	Define master side boundary. Boundary node assignment is made in a counterclockwise direction.
SLBMP p_n p_m	Add a slideline between merged parts p_n and p_m. The surface of p_n along interface represents the slave surface.
SLBP p_n p_m	Add slidelines between adjacent parts p_n and p_m. The surface of part p_n along the interface represents the slave surface. If more than two parts lie along the slideline, this command must be successively repeated as one moves along the interface keeping the slave surface to the left.
SLN *n type* (DYNA2D, NIKE2D)	Define slideline *n* of *type*: EQ. 1: Sliding only EQ. 2: Tied EQ. 3: Frictionless sliding with voids
SLN *n* 4 μ_s μ_k β (DYNA2D)	Define slideline *n* of type 4: frictional sliding with voids. The static coefficient of friction is μ_s. The dynamic coefficient of friction is μ_k. The exponential friction decay constant is β.
SLN *n* 4 *f* (NIKE2D)	Define slideline *n* of type 4: frictional sliding with voids. The static coefficient of friction is μ_s.
SLN *n* 5 r_{tail} z_{tail} r_{head} z_{head} (DYNA2D)	Define slideline *n* of type 5: stone wall. The master surface is defined by any normal vector originating on the wall at (r_{tail},z_{tail}) and terminating at (r_{head},z_{head}), and is assumed to be flat and infinite in extent.
SLN *n* 5 *f* ε^p_{break} (NIKE2D)	Define slideline *n* of type 5: tie breaking. The coefficient of friction *f* is defined for this slideline. Strain ε^p_{break} denotes plastic strain to fail tiebreak.

SLN *n* 6 *f* (DYNA2D, NIKE2D)	Define slideline *n* of type 6: single surface contact. Single surface contact prevents interpenetration between portions of slave surface. A master surface definition is not required. The static coefficient of friction is μ_s.
SLN *n* 7 P_e (NIKE2D)	Define slideline *n* of type 7: eroding pressure contact with separation. This slideline is used in conjunction with auto rezoning to erode material under an erosion pressure P_e.
SLN *n* 7 *q r* (TOPAZ)	Define slideline *n* of type 7: thermal. Conductance *q* and radiation factor $r = \sigma\varepsilon F$ are required for this slideline.
SLN *n* 8 *lc* (NIKE2D)	Define slideline *n* of type 8: merge/release and sliding with separation. This slideline allows merged slideline nodes to be released according to load curve *lc*. Load curve gives number of nodes released as a function of time (useful for fracture).
SLNA Θ_1° Θ_2° (DYNA2D)	Add slideline extensions to beginning and end of master surface of DYNA2D slidelines type 1: sliding only, and type 3: frictionless sliding with voids. Angles are measured counterclockwise from horizontal axis. Extensions default to directions tangent to first and last master segments. When moving along the master surface such that the slave side lies to the left, Θ_1° is extension at the beginning of the master and Θ_2° is extension at the end.
SLV *m n* SLVR *r* SLVS *s*	Define slave side boundary. Boundary node assignment is made in a counterclockwise direction.
SLVM *material_number* SLVN *m n* SLVP *part_number* (DYNA2D)	Set all nodes of *material_number,* nodes *m* through *n* (inclusive), or all nodes of *part_number* as slave nodes for DYNA2D slideline type 5: stonewall. DYNA2D slideline type 5 and command P must be invoked prior to using command SLVM.
SMNO *offset* (DYNA2D)	Add *offset* to all nodes specified with command SLVN. DYNA2D slideline type 5 and command P must be invoked prior to using command SMNO. Command SMNO must be invoked prior to using command SLVN.

3.8.9 Slideline Control

ATN *tolerance* (NIKE2D)	Set Lagrange augmentation *tolerance* in normal direction. Default: 0.0
ATT *tolerance* (NIKE2D)	Set Lagrange augmentation *tolerance* in tangential direction. Default: 0.0
IPF *flag* (NIKE2D)	Set interface penetration *flag* for NIKE2D slideline type 3: frictionless sliding with separation and type 4: frictional sliding with separation with voids. EQ. 0: none (Default) EQ. #: interference load curve number
LAF *flag* (NIKE2D)	Set Lagrange augmentation *flag*: LT. 0: fixed number of augmentations EQ. 0: no augmentations (Default) EQ. 1: augmentations with convergence (multiplier) EQ. 2: augmentations with convergence (gap)
MSDF *flag* (NIKE2D)	Set master surface description *flag*: EQ. 0: surface not smoothed for contact (Default) EQ. 1: surface smoothed for contact
SLFS $\varepsilon_{failure}$ (NIKE2D)	Set slideline failure strain, $\varepsilon_{failure}$. Default: 0.0
SLNEXT on | off (DYNA2D)	Enable / disable slideline extension bypass option. Parameters “on” and “off” must be explictly entered.
SLNI *n m* (DYNA2D)	Slideline *n* intersects slideline *m*. Intersection data applies only to sliding, tied, and frictionless sliding with voids.
SLNP *factor*	Assign a penalty function scale *factor* to most recently defined slideline. This command applies to all slideline types permissible in NIKE2D but only frictional sliding with voids (type 4) in DYNA2D. Default values of 0.1 (DYNA2D) and 1.0 (NIKE2D) are recommended. This command must be preceeded by command SLN.
SLNS *tolerance* (DYNA2D)	Set tolerance for determining initial gap.
SPF *flag* (NIKE2D)	Set small penetration *flag*: EQ. 0: all penetrations considered (Default) EQ. 1: only small penetrations considered
SSDF *flag* (NIKE2D)	Set slave surface description *flag*: EQ. 0: surface not smoothed for constact (Default) EQ. 1: surface smoothed for contact

3.8.10 Explosives: DYNA2D

BDET *m n* RDET *r* SDET *s*	Assign detonation line boundary condition. Detonation nodes are assigned in counterclockwise direction. Commands must be preceeded by command B.
BLAST *option id* $value_1$ $value_2$	Set blast firing parameters needed to calculate pressure delay times on subsequent pressure boundary definitions. Lookup table *id* contains pressure front location verses time. *id* must be defined prior to invoking command BLAST (see command LUT). Delay times are linearly interpolated from table using segment distance from reference plane. *option* parameter defines reference plane as *r* = *value* (*option* = 1) or *z* = *value* (*option* = 2) -- (See table below). This command must be invoked prior to calling commands PBC, PBCR, or PBCS.
DECAY α *n reference_distance*	Set decay multiplier parameters of load curve: *multiplier* $= \alpha$ (*nodal_distance* / *reference_distance*)^n with respect to reference node distance and power (0, 1, 2, or 3) *n* of equation. This command must be invoked prior to calling commands PBC, PBCR, or PBCS.
DETP *n* t_l *m* DETC *side* t_l *m*	Assign nodal point *n* or all nodes on *side* to be located at a detonation point lit at time t_l. If material number $m = 0$ all high explosive materials are lit. If $m \neq 0$ only material *m* is lit. Commands must be preceeded by command B.
GUN *option id* $value_1$	Set gun firing parameters needed to calculate pressure delay times on subsequent pressure boundary definitions. Lookup table *id* contains pressure front location, e.g., projectile's aft, versus time. *id* must be defined prior to invoking command GUN (see command LUT). Delay times are linearly interpolated from table using segment distance from reference plane. *option* parameter defines reference plane as *r* = *value* (*option* = 1) or *z* = *value* (*option* = 2). (See table below). This command must be invoked prior to calling PBC, PBCR, or PBCS.
LDET *m* n_1 ... n_m	Establish detonation line consisting of *m* nodal points n_1 ... n_m. This command must be preceeded by command B.
LUT *id n* d_1 t_1 ... d_n t_n	Define lookup table *id* containing *n* distances and times for establishing pressure boundary curve parameters. All distances and times not contained in the table are linearly interpolated. All times reported on pressure boundary curve are less than zero. This command must be invoked prior to calling commands PBC, PBCR, or PBCS.

RSHAD *r* SHAD *m n* SSHAD s	Establish a shadow boundary. Shadow boundary nodes are assigned in a counterclockwise direction. Detonation points must be defined with command DETP. Command must be preceeded by command B.
SDVEL *vos vis*	Establish detonation velocity of high explosive outside shadow region *vos* and within shadow region *vis*. This command should be used only with the Huygens option.

The following table describes the acceptable specifications to be provided for $value_1$ and $value_2$ with the available *option* of command BLAST and command GUN:

Blast and Gun Option and Value Table

option	Setting of $value_1$	Setting of $value_2$
0	disable	---
1	*r* reference coordinate (planar)	---
2	*z* reference coordinate (planar)	---
3	slope of reference line (linear)	y intercept of reference line (linear)
4	*r* reference coordinate (point)	*z* reference coordinate (point)

3.8.11 Arbitrary Lagrangian-Eulerian Formulations: DYNA2D

ALE / ENDALE	Initiate / terminate arbitrary Lagrangian-Eulerian (ALE) material formulation sequence. ALE functions are defined in the table below.
ABS *begin end material type* ABSR *region type* ABSS *side type*	Set ALE boundary segments for individual material boundary segments. Each boundary segment consists of a beginning node *begin* and ending node *end* defined in counterclockwise order, material number *material* to which these nodes are associated, and boundary relaxation *type*. If *type* is not specified, the default (no relaxation) will be invoked. EQ. 0: No relaxation (Default) EQ. 1: Equal spacing EQ. 2: Normal projection from interior EQ. 3: R-constraint EQ. 4: Z-constraint EQ. 5: Equipotential (symmetric reflection) EQ. 6: Equipotential (antisymmetric reflection) EQ. 7: Proportional to interior Commands ABS, ABSR, and ABSS must be preceeded by command P. These commands establish boundary conditions and are NOT members of the following table: “Arbitrary Lagrangian-Eulerian Formulation Sequences” table.

The following table describes the Arbitrary Lagrangian-Eulerian formulation sequences:

Arbitrary Lagrangian-Eulerian Formulation Sequences

Formulation Command	Definition	Default
matn *n*	DYNA2D material number	0
nstr *steps*	number of time steps between remap	0
mat *type*	material advection type	0
vat *type*	velocity advection type	0
rwf *factor*	remap weighting factor	0
beo *option*	boundary extrapolation option	0
mri *relaxation*	mesh relaxation at initialization	0
mm *n* mat_1 ... mat_n	number of merged materials	0

Arbitrary Lagrangian-Eulerian Formulation Sequences

brf *flag*	boundary relaxation flag	0
stencil *factor*	stencil combination factor	0.0
rsf *factor*	remap scaling factor	0.0
angle Θ°	angle criterion (degrees)	0.0
area *area*	area criterion	0.0
start *time*	ALE start time for material	0.0
endtime *time*	ALE end time for material	0.0
mdt *time*	material death time	0.0
vl *limit*	volume limit	0.0
mind *increment*	minimum displacement increment	0.0
maxd *increment*	maximum displacement increment	0.0

3.8.12 Nodal Constraints: DYNA2D - NIKE2D

CNP *cnode m n i*
CNPB *cnode s i*

Establish a constrained nodal pair. Constraining node *cnode* denotes the first node of the constrained nodal pair. The boundary nodes must be defined in counterclockwise order. The direction of constraint *i*:
EQ. 1: radial constraint
EQ. 2: axial constraint
EQ. 3: radial and axial constraints

The following table describes the actions taken by the constrained nodal pair commands CNP and CNPB when duplicate constrained nodal pairs are entered:

Constrained Nodal Pair Boundary Condition Action Table

Duplicate Constrained Node Direction	Original Constrained Node Direction	Action
1	1	Duplicate nodal pair is not entered into list. Original nodal pair is retained.
1	2	Duplicate nodal pair is entered into list.
1	3	Duplicate nodal pair is not entered into list. Original nodal pair is retained.
2	1	Duplicate nodal pair is entered into list.

Constrained Nodal Pair Boundary Condition Action Table

2	2	Duplicate nodal pair is not entered into list. Original nodal pair is retained.
2	3	Duplicate nodal pair is not entered into list. Original nodal pair is retained.
3	1, 2, 3	All previous instances of nodal pairs containing direction codes: 1, 2, 3 are deleted from list. Duplicate nodal pair is entered into list.

3.8.13 J-Integral: NIKE2D

JCOORD c_x c_z

Set x- and z-components (c_x,c_z) of the crack tip location. This command must be preceeded by command JINT.

JCT *m n*
JCTC *corner*
JCTR *r*
JCTS *side*

Include boundary nodes in crack tip definition. Commands must be preceeded by commands JINT and B.

JINT *contours* p_x p_z

Establishment of J-Integral mode. Set number of contours contours to be evaluated and the x- and z-components (p_x,p_z) of the vector pointing in the direction of crack advancement. This command must be invoked first in the sequence of all subsequent J-Integral commands.

JPHASE *plus minus larrot*

Set mode mixity separation option for homogeneous isotropic elastic or interfacial isotropic elastic cracks. Material number *plus* denotes the region above crack tip. Material number *minus* denotes the region below crack tip. Option *larrot* specifies compensation for large rigid-body crack tip rotations during mode separation:
EQ. 0: No adjustments are made (Default)
EQ. 1: Compensation for large rigid-body crack tip rotations is made
This command must be preceeded by command JINT.

JTHERM *option*

Set J-Integral thermal *option*:
EQ. 0: thermal option off (Default)
EQ. 1: thermal option on
This command must be preceeded by command JINT and command B.

3.8.14 Boundary Conditions: TOPAZ

CBC *m n k* r_1 r_2 *j* r_3 r_4
CBCR *r k* r_1 r_2 *j* r_3 r_4
CBCS *s k* r_1 r_2 *j* r_3 r_4

Assign a convection boundary condition. The convection boundary condition is assigned in counterclockwise direction along boundary. Fluid temperature (T_∞) is defined as a function of time by load curve *k*. Fluid temperature is scaled along boundary using r_1 and r_2 as temperature multipliers associated with first and last boundary nodes, respectively. For a fluid temperature which does not vary with a load curve, set: $k = 0$, and r_1 and r_2 as temperature values. The convective heat transfer coefficient is defined with a time dependent, positive load curve *j* and can be scaled using r_3. The coefficient can also be scaled in proportion to temperature difference between boundary temperature and fluid temperature using r_4 as an exponent, i.e., $(T_{boundary} - T_{fluid})^{r4}$. Alternatively, for a convective heat transfer coefficient which does not vary with a load curve, set $j = 0$ and r_3 as the coefficient value.

FBC *m n k* r_1 r_2
FBCR *r k* r_1 r_2
FBCS *s k* r_1 r_2

Assign a flux load boundary condition varying in time according to load curve *k*. Time dependent, temperature dependent, and constant functional relations are indicated by positive, negative, and zero load curve numbers, respectively. Flux curve multipliers r_1 and r_2 correspond to the first node and second node, respectively. The flux vector requires a negative sign for transfer into the part.

RBC *m n k* r_1 r_2 *j* r_3
RBCR *r k* r_1 r_2 *j* r_3
RBCS *s k* r_1 r_2 *j* r_3

Assign a radiation boundary condition in counterclockwise direction along boundary. The temperature of the surroundings (T_∞) is a function of time defined by load curve *k*. This temperature can be scaled along boundary using temperature multipliers r_1 and r_2 associated with first and last node numbers, respectively. For a surrounding temperature which does not vary with a load curve, set: $k = 0$, and r_1 and r_2 as temperature values. The radiation heat transfer coefficient is defined with a time dependent, positive load curve, *j*, or a temperature dependent, negative load curve, *-j*. The radiation heat transfer coefficient is the product of Stefan-Boltzmann constant, total hemispherical surface emmissivity, and view factor between surface and the surroundings ($\sigma\varepsilon F$) and can be scaled using r_3. For a radiation heat transfer coefficient which does not vary with a load curve, set: $j = 0$, and r_3 as the coefficient value.

TBC *m n k r*
TBCR *r k r*
TBCS *s k r*

Assign a temperature boundary condition varying in time according to load curve *k* and multiplied by *r.* Time dependent, temperature dependent, and constant functional relations are indicated by positive, negative, and zero load curve numbers, respectively. The constant value will be the curve multiplier *r.*

T0 *temperature*

Assign initial / reference temperature of all nodes to temperature.

3.8.15 Element Heat Generation: TOPAZ

EGR *m n k r*

Assign an element heat generation boundary condition to elements *m* through *n*. Volumetric heat generation loads vary in time according to a load curve *k* and a load curve multiplier *r.* Time dependent, temperature dependent, and constant functional relations are indicated by positive, negative, and zero load curve numbers, respectively. For element heat generation which does not vary with a load curve, set $k = 0$ and r to the value of volumetric heat generation.

3.8.16 Enclosure Radiation: TOPAZ

EBC *m n k l* r_1
EBCR *r k l* r_1
EBCS *s k l* r_1

Assign an enclosure radiation boundary condition. The boundary condition is assigned in a counterclockwise direction along the boundary. For a segment with absorption and reflection, set: $k = 0$. For a segment which allows only absorption, e.g., a hole in a cavity, set $k = 1$. The surface emissivity is defined as a funciton of wavelength with a load curve *l*. This load curve is defined with commands ECD and LAMBDA. Setting $l = 0$ establishes a surface emissivity of 1.0 and employs r_1 as the segment temperature. Otherwise, r_1 is not used. Command ERCC is also used in the implementation of enclosure radiation.

ECD *n* e_1 ... e_n

Establish emissivity curve definition *n* composed of emissivity values e_1 ... e_n commensurate with the number of wavelength breakpoints defined by command LAMBDA. Command LAMBDA must be called prior to invoking command ECD.

ERCC *u c r s*	Establish enclosure ratiation control card parameters: *u* = K: Kelvin temperature units *u* = R: Rankine temperature units *c*: Stefan-Boltzmann constant *r*: radiosity convergence tolerance *s*: maximum number of radiosity iterations If the number of wavelength breakpoints (see command LAMBDA) is > 1, use format: ERCC *u c r s* otherwise use format: ERCC *u c*
LAMBDA *n* λ_1 ... λ_n	Establish number *n* wavelength breakpoints λ_1 ... λ_n.

3.8.17 Miscellaneous Boundary Conditions

MBCS	Write miscellaneous boundary conditions on all sides of all parts to mazout file.

3.8.18 DYNA2D Control

DYNA2D Card #1

TITLE	Define problem title in MAZE output file. The title must be placed on the next line of the MAZE input file.

DYNA2D Card #2

INPSD *ndmat numeld nummas*	Set discrete springs, dampers, and masses containing *ndmat* material definitions for discrete elements, *numeld* discrete springs and dampers, and *nummas* discrete masses. This command must be followed by: 1. *ndmat* lines for each discrete element material control card containing: discrete element material number, material type, and material data; 2. *numeld* lines for each discrete element control card containing: discrete element number, first node, second node, discrete element material number, generation increment, scale factor on computer force; and 3. *nummas* lines for each lumped mass data card containing: node number, mass.
BFGR *k s* BFGZ k s	Set body force load due to base acceleration in *r*-direction (BFGR) or *z*-direction (BFGZ). Load is based upon the base acceleration time history as defined in load curve *k* subject to scale factor *s*. This command is applicable to plane strain geometries only. Default: 0.0
BFSX *k s*	Set body force load due to angular velocity about *x*-axis. Load is based upon velocity time history as defined in load curve *k* subject to scale factor *s*. Angular velocity is measured in units of radians per unit time. Default: 0.0
BFSZ *k s*	Set body force load due to angular velocity about *z*-axis. Load is based upon the velocity time history as defined in load curve *k* subject to scale factor *s*. Angular velocity is measured in units of radians per unit time. Default: 0.0

DYNA2D Card #4

IAUTO *option n* m_1 ... m_n	Set *option* for automatic contact of materials: EQ. 0: no automatic contact (Default) EQ. 1: active material list to follow: number of materials *n* and each material number m_n must be defined. EQ. 2: all materials are active for automatic contact
SCS *scope*	Set *scope* of contact searching: EQ. 0: (Default) EQ. 1: fastest search EQ. 4: most robust search
SFAS *frequency*	Set search *frequency* for automatic slidelines. Default: 0
FASP *scale*	Set *scale* factor for automatic slideline penalty number. Default: 0
LVFC μ_s	Set low velocity friction coefficient μ_s. Default: 0.0
HVFC μ_k	Set high velocity friction coefficient μ_k. Default: 0.0
EFDC β	Set exponential friction decay constant β. Default: 0.0

DYNA2D Card #5

TERM *time*	Terminate calculation at *time*. This command may be used instead of command DELT. Default: 0.0
ITSS Δt_0	Set initial time step size. Default: 0.0 -- an initial time step size is computed.
SBRF *steps*	Set number of time *steps* between restart dumps. Default: 0
TSSF *scale*	Set *scale* factor for computed time step size. Default: 0.667
RFMTS *factor*	Set reduction *factor* to determine minimum permissible time step. Default: 0.0
REZO *begin end between*	Set time to *begin*, *end*, and time *between* automatic re-zoning. Default: 0.0

DYNA2D Card #6

PRTI *time_interval*	Set node and element dump *time interval* for printing.
PLTI *time_interval*	Set node and element dump *time interval* for ORION.
PRTT $\Delta time_{print}$	Set node and element dump time interval for printing.
PLTT $\Delta time_{plot}$	Set node and element dump time interval for ORION.
NPBK $m\ n^1_1\ n^1_2 \ldots n^m_1\ n^m_2$	Write nodal data for *m* blocks into the MAZE output file. The k^{th} block will include all nodes n^k_1 to n^k_2.
EPBK $m\ e^1_1\ e^1_2 \ldots e^m_1\ e^m_2$	Write element data for *m* blocks into the MAZE output file. The k^{th} block will include all nodes e^k_1 to e^k_2.
IED *option*	Set output *option* for internal energy: EQ. 0: internal energy is not written into state plot database EQ. 1: internal energy is written into state plot database (Default)
SDO *option*	Set *option* for chemistry dump output: EQ. 0: p^2T is written into state plot database (Default) EQ. 1: temperature is written into state plot database
HVDF *flag*	Set history variable dump *flag*: EQ. 0: history variables are not written into state plot database (Default) EQ. 1: history variables are written into state plot database
PVDF *flag*	Set peak value dump *flag*: EQ. 0: peak values are not written into state plot database (Default) EQ. 1: peak values of displacements, velocities, max/min principal stress, and max/min pressure are written into state plot database for each element
TIBU $\Delta time_{display\ updates}$	Set *time interval* between updates of analysis display. Default: 0.0
TSBS steps	Number of time *steps* between status updates print to file. Default: 0

DYNA2D Card #7

IGM *type*	Set geometry *type*: EQ. 0: axisymmetric (Default) EQ. 1: plane strain
BRODE	Initiate Brode function sequence. (See table below.)
ENDBRODE	Terminate Brode function sequence.
GRVS *a n* ρ_1 z_1 ... ρ_n z_n	Set gravity stress initialization for *n* points of mass density ρ as a function of depth *z* to acceleration *a*.
TEO *option*	Set thermal effects *option*: EQ. 0: no thermal effects EQ. -1: new temperature state is read from file for each time step. Time word at the beginning of each temperature is ignored. EQ. -2: temperature state is interpolated from temperature states in file. Time word at the beginning of each temperature is used. EQ. -3: file contains only one temperature state. The initial state is 0.0.

The following table describes the functions available in the Brode function sequence:

Brode Function Sequence

Function Command	Definition	Default
yield	yield (Ktons)	0.0
height	height of burst	0.0
x	DYNA2D x-coordinate of Brode origin	0.0
y	DYNA2D y-coordinate of Brode origin	0.0
z	DYNA2D z-coordinate of Brode origin	0.0
time	initiation time for Brode function	0.0
lctoa	optional load curve number giving TOA shift vs. range	0
lcyld	optional load curve number giving yield vs. time for scaling calculations	0
cfftl	conversion factor: feet to DYNA2D length units	0.0
cfmst	conversion factor: ms to DYAN2D time units	0.0
cfpsi	conversion factor: psi to DYNA2D pressure units	0.0

DYNA2D Card #8

DHQT *method*	Hourglass stabilization *method*: EQ. 0: Default EQ. 1: Standard DYNA2D EQ. 2: Rotational EQ. 3: Flanagan-Belytschko EQ. 4: Hancock EQ. 5: stiffness
DHGQ Q_h	Hourglass viscosity coefficient (Default: $Q_h = 0$): EQ. 1: $Q_h \leq 0.15$ for stability EQ. 2: $Q_h \leq 0.20$ for stability EQ. 3: $Q_h \leq 0.40$ for stability EQ. 4: $Q_h \leq 0.40$ for stability
DBQT *type*	Bulk viscosity *type*: EQ. 0: Default EQ. 1: Standard DYNA2D bulk viscosity EQ. 2: Richards-Wilkins bulk viscosity
DQQ Q_q	Quadratic shock viscosity coefficient (Default: $Q_q = 1.5$)
DQL Q_l	Linear shock viscosity coefficient (Default: $Q_l = 0.06$)
SRDR *rate*	Stress *rate* default reset: EQ. 0: DYNA2D default stress rate for this material EQ. 1: Jaumann rate EQ. 2: Green-Naghdi rate

3.8.19 NIKE2D Control

NIKE2D Card #1

TITLE — Define problem title in MAZE output file. The title must be placed on the next line of the MAZE input file.

NIKE2D Card #2

NPBK $m\ n^1_1\ n^1_2 \ldots n^m_1\ n^m_2$ — Write nodal data for m blocks into the MAZE output file. The k^{th} block will include all nodes n^k_1 to n^k_2.

EPBK $m\ e^1_1\ e^1_2 \ldots e^m_1\ e^m_2$ — Write element data for m blocks into the MAZE output file. The k^{th} block will include all nodes e^k_1 to e^k_2.

NIKE2D Card #3

BFGR *k s*
BFGZ k s — Set body force load due to base acceleration in *r*-direction (BFGR) or *z*-direction (BFGZ). Load is based upon the base acceleration time history as defined in load curve *k* subject to scale factor *s*. This command is applicable to plane strain geometries only. Default: 0.0

BFSZ *k s* — Set body force load due to angular velocity about *z*-axis. Load is based upon the velocity time history as defined in load curve *k* subject to scale factor *s*. Angular velocity is measured in units of radians per unit time. Default: 0.0

NCNM *n* — Set number of concentrated nodal masses. Default: 0

NCND *n* — Set number of concentrated nodal dampers. Default: 0

NIKE2D Card #4

SMOPT *flag* — Set element formulation *flag*:
EQ. 0: B-bar
EQ. 1: NIKE2D formulation (Default)

NEIP *flag* — Set integration order *flag*:
EQ. 0: 2×2 Gauss integration
EQ. 1: 1 point integration

NBFL *n* — Set *n*umber of element body forces. Default: 0

TEO *option*	Set thermal effects *option*: EQ. 0: no thermal effects EQ. 1: nodal temperatures are spatially invariant and are determined from load curve command ITCURV at each time step. EQ. 2: nodal temperatures are determined from specified vectors $\mathbf{T}^{\mathbf{B}}$ and $\mathbf{T}^{\mathbf{M}}$ and from load curve command ITCURV at each time step. EQ. -1: nodal temperatures are determined by reading in a new temperature state from a TOPAZ2D plotfile at each time step. The time word at the beginning of each temperature is ignored. EQ. -2: nodal temperatures are interpolated from the temperature states in a TOPAZ2D plotfile. EQ. -3: initial nodal temperatures are defined at $\mathbf{T}^{\mathbf{R}}$ and final nodal temperatures are determined from a steady-state TOPAZ2D plotfile. At each intermediate step, nodal temperatures are interpolated from these two states.
ITCURV *load_curve*	Set *load_curve* number for temperature vs. time.
ITRF *flag*	Set initial temperature reference *flag*: EQ. 0: nodal reference temperatures are not specified. EQ. 1: nodal reference temperatures $\mathbf{T}^{\mathbf{R}}$ are specified on node cards.
IEPD *flag*	Set element plot database *flag*: EQ. 0: element energy is contained in the plot database EQ. 1: element thickness (plane stress) is contained in the plot database. EQ. 2: element temperature is contained in the plot database

NIKE2D Card #5

IGM *type*	Set geometry *type*: EQ. 0: axisymmetric (Default) EQ. 1: plane strain EQ. 2: plane stress
AF *flag*	Set analysis *flag*: EQ. -2: dynamic analysis, statically initialized EQ. -1: dynamic analysis EQ. 0: quasi-static analysis EQ. *n*: Perform eigenvalue analysis with *n* eigenvalues and eigenvectors to be extracted. See command NEIG.
NEIG *n*	Perform eigenvalue analysis to obtain the first *n* eigenvalues. See command AF.
BWMO *flag*	Set bandwidth minimization *flag*: EQ. 0: no bandwidth minimization EQ. 1: perform bandwidth minimization (Default)
IOOSF *flag*	Set out-of-core solution *flag*: EQ. 0: perform in-core solution (Default) EQ. 1: perform out-of-core solution
PCM *percent*	Set percentage of computer memory to be used.
SM *method*	Set solution *method* to be used: EQ. 0: fixed step strategy EQ. 1: adaptive ISLAND strategy
NIP1 γ	Set Newmark parameter γ. Default: 0.5
NIP2 β	Set Newmark parameter β. Default: 0.25
TSSF *scale*	Set *scale* factor for computed time step size. Default: 0.667

3.8.20 NIKE2D Solution Definition

NIKE2D Solution Card #1

DELT $\Delta time$	Set time step size.
NSTEP *steps*	Set number of time *steps*.
PRTI *step_interval*	Set node and element dump *step interval* for printing.
PLTI *step_interval*	Set node and element dump *step interval* for ORION.
PRTT $\Delta time_{print}$	Set node and element dump time interval for printing.
PLTT $\Delta time_{plot}$	Set node and element dump time interval for ORION.
SBRF *steps*	Set number of time *steps* between restart dumps.
SIAR *interval*	Set step *interval* for automatic rezoning. Default: 0
MSRF *reformations*	Set maximum number of stiffness matrix *reformations* per time step.
NSMD *flag*	Set standard solution method *flag*: (Default: 0) LE. 1: BFGS EQ. 2: Broyden EQ. 3: Davidon-Fletcher-Powell (DFP) EQ. 4: Davidon symmetric EQ. 5: modified Newton EQ. 6: arc length EQ. 7: arc length with line search EQ. 8: arc length with BFGS EQ. 9: arc length with Broyden EQ. 10: arc length with DFP EQ. 11: arc length with modified BFGS EQ. 12: arc length with Davidon
DCTOL *tolerance*	Set convergence *tolerance* on displacements. Default: 0.001
ECTOL *tolerance*	Set convergence *tolerance* on energy. Default: 0.01
NBSR *steps*	Set number of time *steps* between stiffness matrix reformations. Default: 0 -- set to 1
NBEI *steps*	Set number of time *steps* between equilibrium iterations. Default: 0 -- set to 1
NIBSR *iterations*	Set maximum number of equilibrium *iterations* allowed between stiffness matrix reformations.

NIKE2D Solution Card #2

NAUS *steps*	Set number of arc length unloading *steps*. Default: 0
IAUM *method*	Set arc length unloading *method*: (Default: 0) LE. 1: BFGS EQ. 2: Broyden EQ. 3: Davidon-Fletcher-Powell (DFP) EQ. 4: Davidon symmetric EQ. 5: modified Newton
IADC *flag*	Set arc length displacement control *flag:* EQ. 0: displacement norm is used (Default) GE. 0: node number for arc length displacement control
IADR *direction*	Set *direction* for nodal arc length displacement control: EQ. 1: *r* direction EQ. 2: *z* direction
IACN *method*	Set arc length constraint *method:* EQ. 0: Crisfield (Default) EQ. 1: Ramm
IADM *flag*	Set arc length damping *flag*: EQ. 0: arc length damping is specified (Default) EQ. 1: arc length damping is not specified
ASIZ *size*	Set initial arc length *size*: EQ. 0: arc length based upon time step size (Default)

NIKE2D Solution Card #3

LST *tolerance*	Set line search *tolerance*: [0.5 ≤ *tolerance* ≤ 0.95]. Default: 0.9
SST *tolerance*	Set slideline stiffness insertion *tolerance*. Default: 0.001
RFFC *factor*	Set reduction *factor* for frictional slideline. Default: 0.01
RLT *tolerance*	Set rezoner least squares fit *tolerance*. Default: 0.01
IGS *flag*	Set geometric stiffness *flag*: EQ. 0: set geometric stiffness flag (Default) EQ. 1: do not set geometric stiffness flag

3.8.21 NIKE2D ISLAND Template Commands

DCTOL *tolerance*	Set convergence *tolerance* on displacements. Default: 0.001
DELTA *size* *time initial*	Set initial time step *size.* Default: 0.0
DSTOL *tolerance*	Set step displacement *tolerance*. Default: 10.0
DTMAX *size* *maximum time step*	Set maximum time step *size*. Default: 0.0
DTMIN *size* *minimum time step*	Set minimum time step *size*. Default: 0.0
ECTOL *tolerance*	Set convergence *tolerance* on energy. Default: 0.01
MAXSTEPS *steps*	Set maximum number of time *steps*. Default: 0
MAXTRIES *changes*	Set maximum number of time step size *changes*. Default: 10
MSRF *reformations*	Set maximum number of stiffness matrix *reformations* per time step. Default: 3
NGOODSTEPS *steps*	Set number of time *steps* for time step size changes. Default: 2
NIBSR *iterations*	Set maximum number of equilibrium *iterations* allowed between stiffness matrix reformations. Default: 10
NUMREF *reformations*	Set number of *reformations* for a good step. Default: 3
RCTOL *tolerance*	Set *tolerance* on residuals. Default: 10.0
SBRF *steps*	Set number of time *steps* between restart dumps.
TERM *time*	Terminate calculation at *time*. Default: 0.0
TSSF *scale*	Set *scale* factor for computed time step size. Default: 0.667

3.8.22 TOPAZ2D Control Commands

TOPAZ2D Card #1

TITLE — Define title of MAZE output file. The title must be placed on the next line of MAZE input file.

TOPAZ2D Card #2

IGM *type* — Set geometry *type*:
EQ. 1: axisymmetric (Default)
EQ. 2: plane

BWMO *flag* — Set bandwidth minimization *flag*:
EQ. 0: do not perform bandwidth minimization (Default)
EQ. 1: perform bandwidth minimization
EQ. 2: nodal destination vector read from input file

NSMD *method* — Set solution *method*:
EQ. 0: out-of-core profile solver (Default)
EQ. 1: in-core profile solver
EQ. 2: diagonally scaled conjugate gradient
EQ. 3: incomplete Cholesky conjugate gradient
EQ. 4: nonsymmetric profile solver

CGCTOL *tolerance* — Set conjugate gradient convergence *tolerance*:
EQ. 0: tolerance = 1000.0 × machine roundoff (Default)
Suggested range of tolerance: 10^{-4} to 10^{-6}.

TOPAZ2D Card #3

RTYPE *type* — Set radiation calculation *type*. Positive *type:* view factor or exchange factor matrix will be read from file *vfl*; negative *type:* view factor or exchange matrix will be read from current input source:
EQ. 0: view factors (Default)
EQ. 1: exchange factors
EQ. 2: black body exchange factors

TOPAZ2D Card #4

ANALYSIS *type*	Set analysis *type*: EQ. 0: steady state (Default) EQ. 1: transient with consistent mass matrix EQ. 2: transient with lumped mass matrix
STEP *code*	Set time step *code*: EQ. 0: fixed time step (Default) EQ. 1: variable time step
PRTI *step_interval*	Set node and element dump *step interval* for printing.
PLTI *step_interval*	Set node and element dump *step interval* for ORION.
PRTT $\Delta time_{print}$	Set node and element dump time interval for printing.
PLTT $\Delta time_{plot}$	Set node and element dump time interval for ORION.
SBRF *steps*	Set number of time *steps* between restart dumps.
ALPHA γ	Set Newmark parameter γ: EQ. 0.5: Crank Nicolson (Default) EQ. 1.0: fully implicit

TOPAZ2D Card #5: Fixed Time Step

START $time_{initial}$	Set initial problem time. Default: 0.0
TERM $time_{final}$	Set termination problem time. Default: 0.0
DELTA *size*	Set time step *size.* Default: 0.0
NSTEP *steps*	Set number of *steps* between initial and final problem time.

TOPAZ2D Card #5: Variable Time Step

START $time_{initial}$	Set initial problem *time*. Default: 0.0
TERM $time_{final}$	Set termination problem *time*. Default: 0.0
DELTA $size_{time\ initial}$	Set initial time step *size.* Default: 0.0
NSTEP *steps*	Set number of *steps* between initial and final problem time.
DTMIN $size_{minimum\ time\ step}$	Set minimum time step *size*.
DTMAX $size_{maximum\ time\ step}$	Set maximum time step *size*.

TMPMAX *temperature*
Set maximum *temperature* change in each time step above which the time step will be decreased. Default: 0.0

TSSF *parameter*
Set time step control *parameter.* Range of *parameter*:
$0.0 < parameter < 1.0$
Default: 0.5

TOPAZ2D Card #6

NONL *type*
Set *type* of problem:
EQ. 0: linear problem (Default)
EQ. 1: nonlinear problem -- material properties evaluated at Gauss point temperature
EQ. 2: nonlinear problem -- material properties evaluated at element average temperature

MSRF *reformations*
Set maximum number of coefficient matrix *reformations* per time step.

NIBSR *iterations*
Set maximum number of equilibrium *iterations* allowed between coefficient matrix reformations.

TCTOL *tolerance*
Set convergence *tolerance*.
EQ. 0: tolerance = 1000.0 × machine roundoff (Default)
Suggested range of *tolerance*: 10^{-4} to 10^{-6}.

RELAX *parameter*
Set divergence control *parameter.*
Steady state: $0.3 \leq parameter \leq 1.0$ (Default: 1.0)
Transient state: $0.0 \leq parameter \leq 1.0$ (Default: 0.5)

3.8.23 CHEMICAL TOPAZ2D Control

CHEMICAL TOPAZ2D Card #1

TITLE — Define title of MAZE output file. The title must be placed on the next line of the MAZE input file.

CHEMICAL TOPAZ2D Card #2

IGM *type* — Set geometry *type*:
EQ. 1: axisymmetric (Default)
EQ. 2: plane

BWMO *flag* — Set bandwidth minimization *flag*:
EQ. 0: do not perform bandwidth minimization (Default)
EQ. 1: perform bandwidth minimization
EQ. 2: nodal destination vector read from input file

NSMD *method* — Set solution *method*:
EQ. 0: out-of-core profile solver (Default)
EQ. 1: in-core profile solver
EQ. 2: diagonally scaled conjugate gradient
EQ. 3: incomplete Cholesky conjugate gradient
EQ. 4: nonsymmetric profile solver

CGCTOL *tolerance* — Set conjugate gradient convergence *tolerance*.
EQ. 0: tolerance = 1000.0 × machine roundoff
Suggested range of *tolerance*: 10^{-4} to 10^{-6}.

CHEMICAL TOPAZ2D Card #3

RTYPE *type* — Set radiation calculation *type*. Positive *type*: view factor or exchange factor matrix will be read from file *vfl*; negative *type*: view factor or exchange matrix will be read from current input source:
EQ. 0: view factors (Default)
EQ. 1: exchange factors
EQ. 2: black body exchange factors

CHEMICAL TOPAZ2D Card #4

ANALYSIS *type*	Set analysis *type*: EQ. 0: steady state (Default) EQ. 1: transient with consistent mass matrix EQ. 2: transient with lumped mass matrix
STEP *code*	Set time step *code*: EQ. 0: fixed time step (Default) EQ. 1: variable time step
IPRTI *step_interval*	Set node and element dump *step interval* for printing.
IPLT *step_interval*	Set node and element dump *step interval* for ORION.
SBRF *steps*	Set number of time *steps* between restart dumps.
NIP1 γ	Set Newmark parameter γ: EQ. 0.5: Crank Nicolson (Default) EQ. 1.0: fully implicit
RPRT $\Delta time_{print}$	Set node and element dump *time interval* for printing.
RPLT $\Delta time_{plot}$	Set node and element dump *time interval* for ORION.

CHEMICAL TOPAZ2D Card #5: Fixed Time Step

START *time* $_{initial}$	Set initial problem *time*. Default: 0.0
TERM *time* $_{final}$	Set termination problem *time*. Default: 0.0
DELTA *size*	Set time step *size.* Default: 0.0
NSTEP *interval*	Set number of *interval* between initial and final problem time.

CHEMICAL TOPAZ2D Card #5: Variable Time Step

START *time* $_{initial}$	Set initial problem *time*. Default: 0.0
TERM *time* $_{final}$	Set termination problem *time*. Default: 0.0
DELTA *size* $_{time\ initial}$	Set initial time step *size.* Default: 0.0
NSTEP *interval*	Set number of *interval* between initial and final problem time.
DTMIN *size* $_{minimum\ time\ step}$	Set minimum time step *size*. Default: 0.0
DTMAX *size* $_{maximum\ time\ step}$	Set maximum time step *size*. Default: 0.0

TMPMAX *temperature*	Set maximum *temperature* change in each time step above which the time step will be decreased. Default: 0.0
TSSF *parameter*	Set time step control *parameter.* Range of *parameter*: $0.0 < parameter < 1.0$ (Default: 0.5)

CHEMICAL TOPAZ2D Card #6

NONL *type*	Set *type* of problem*:* EQ. 0: linear problem (Default) EQ. 1: nonlinear problem -- material properties evaluated at Gauss point temperature EQ. 2: nonlinear problem -- material properties evaluated at element average temperature
MSRF *reformations*	Set maximum number of stiffness matrix *reformations* per time step. Default: 1
NIBSR *iterations*	Set maximum number of equilibrium *iterations* allowed between stiffness matrix reformations. Default: 10
TCTOL *tolerance*	Set convergence *tolerance*. EQ. 0: tolerance = 1000.0 × machine roundoff Suggested range of *tolerance*: 10^{-4} to 10^{-6}.
RELAX *parameter*	Set divergence control *parameter.* Steady state: $0.3 \leq parameter \leq 1.0$ (Default: 1.0) Transient state: $0.0 \leq parameter \leq 1.0$ (Default: 0.5)

CHEMICAL TOPAZ2D Chemistry Control Card

NRX *n*	Set *n*umber of chemical reactions.
NMIX *materials*	Set number of mixture *materials*.
MTMD *flag*	Set material property calculation *flag*: EQ. 1: alpha calculation EQ. 2: energy conserving
CHMT *materials*	Set number of *materials* used in chemistry.
GPLC *type*	Set *type* of chemistry composition calculation: EQ. 1: calculated at Gauss points EQ. 2: calculated at element points
CTIN *type*	Set *type* of temporal solution scheme for chemistry: EQ. 1: calculation at alpha time

NRX2 *number*	Set *number* of chemical reactions of type 2.
EUBG *coming_from_material*	Set *"coming from" material* for eff. Eulerian calculation.
EUED *going_to_material*	Set *"going to" material* for eff. Eulerian calculation.
EUAL *tolerance*	Set error *tolerance* for effective Eulerian calculation.
EUAR *area*	Set maximum *area* for reaction front using effective Eulerian calculation.

CHEMICAL TOPAZ2D Reaction Control Card #1

GASS *constant*	Set gas *constant.*
ERRX *tolerance*	Set convergence *tolerance* for chemical reaction rates.
ITRX *iterations*	Set maximum number of Newton-Raphson *iterations* to converge chemical reaction rates.
PACT *flag*	Set pressure active *flag*: EQ. -1: pressure terms used; problem terminates at maximum pressure of reaction EQ. 0: pressure terms ignored EQ. 1: pressure terms used
MPL1 *species*	Set first *species* number to be plotted.
MPL2 *species*	Set second *species* number to be plotted.

CHEMICAL TOPAZ2D Reaction Control Card #2

MINT *temperature*	Set minimum *temperature* of reaction.
MAXT *temperature*	Set maximum *temperature* of reaction.
PON *pressure*	Set initial / final *pressure*.
PMAX *pressure*	Set maximum *pressure* of reaction.

CHEMICAL TOPAZ2D Reaction Data Card #1

REAC	Initiate reaction process. This command MUST be called prior to calling all other commands associated with Chemical Reaction Data.
CTYPE *type*	Set chemical reaction *type*: EQ. 1: standard reaction EQ. 2: equilibrium reaction EQ. 3: effective Eulerian reaction EQ. 4: partial pressure based reaction
RMNT *temperature*	Set minimum *temperature* of reaction.
RMIT *value*	Set minimum *value* of 1/T of reaction..

CHEMICAL TOPAZ2D Reaction Data Card #2

RKK *logarithm*	Set *logarithm* of collision frequency.
EEX *energy*	Set activation *energy* of collision frequency.
PFAC *exponent*	Set pressure prefactor *exponent*.
VEX *volume*	Set activation *volume*.

CHEMICAL TOPAZ2D Reaction Data - Additional Cards

MIXN $n\ p_1 \ldots p_n$	Establish list of reacting materials consisting of n parts composed of reacting materials for parts $p_1 \ldots p_n$.
REAC	Initiate reaction process. This command MUST be called prior to calling all other commands associated with Chemical Reaction Data.
STOC $n\ s_1 \ldots s_n$	Establish list of stoichiometric values consisting of n values composed of stoichometries $s_1 \ldots s_n$.
STOS $m\ n\ s_m \ldots s_n$	Establish initial reaction number m, final reaction number n, and stoichiometries of reactions $s_m \ldots s_n$. $m \le n$.
IPRF $n\ c_1 \ldots c_n$	Establish list of composition exponents of reaction consisting of n exponents for $c_1 \ldots c_n$.
IPRS $m\ n\ c_m \ldots c_n$	Establish initial reaction number m, final reaction number n, and composition exponents of reactions $c_m \ldots c_n$. $m \le n$.

3.9 Transition From Phase II To Phase III

3.9.1 Analysis Code Establishment

WBCD *format*	Prepare disk file containing mesh data written in accordance with *format*, e.g., DYNA2D, NIKE2D, TOPAZ2D specifications required by analysis code.

3.10 Phase III Commands

3.10.1 Graphics Commands

ELPLT	Display element numbers on mesh of materials. This command must be preceeded by command M.
FSON / FSOFF	Enable / disable the display of free surfaces and slideline interfaces by invoking command O. Command FSON must be invoked prior to command O. Default: FSON
G	Display mesh with all material numbers.
GO	Display mesh to the right of centerline and display outline to the left of centerline.
M *material*	Display material in all plots invoked by commands EPLT, MO, NDPLT, and V. This command must be invoked prior to commands EPLT, MO, NDPLT, and V.
MNON / MNOFF	Enable / disable the display of material numbers with commands G, GO, and O. Default: MNON
MO	Display outline of material. This command must be preceeded by command M.
NDPLT	Display node numbers on mesh of material. This command must be preceeded by command M.
O	Display outlines of all materials.
V	Display mesh of material. This command must be preceeded by command M.

3.10.2 General Commands

CMN e_1 e_2 *material*	Change material numbers of elements e_1 through e_2 to *material*.
CSF *factor*	Scale nodal coordinates by scale *factor.*
CSHF Δr Δz	Translate nodal coordinates by (Δr,Δz).
NEOS *node_offset element_offset*	Establish node and element offset. Node numbers in MAZE output file will begin at node *node_offset* + 1; element numbers in MAZE output file will begin at element *element_offset* + 1.
PHS2	Return to Phase II command section of MAZE.

3.10.3 Initial Nodal Temperatures: NIKE2D - TOPAZ

ERIT *first last step temperature*	Assign initial / reference temperature of all nodes associated with elements in the range from *first* through *last* inclusive by step size to *temperature*.
MRIT *material temperature*	Assign initial / reference temperature of all nodes in material to *temperature*.
NRIT *first last step temperature*	Assign initial / reference temperature of all nodes in the range from *first* through *last* inclusive by step size to *temperature*.
T0 *temperature*	Assign initial / reference temperature of all nodes to *temperature*.
TIC *m n temperature*	Assign an initial temperature condition to nodes *m* through *n*.
TICV r_1 z_1 r_2 z_2 r_3 z_3 r_4 z_4 *temperature*	Assign an initial temperature boundary condition to the quadrilaterial bounded by coordinates (r_1,z_1) to (r_4,z_4). The coordinates of the quadrilaterial must be specified in a counterclockwise direction. Temperature loads are assigned in a counterclockwise direction. This command must be preceeded by command B.

3.10.4 Material Commands

DYNA2D and NIKE2D material definitions begin with command MAT and are followed by material type-dependent commands. The material definition is completed with command ENDMAT. If a DYNA2D material requires an equation-of-state, the material definition must immediately be followed by command EOS. All relevant equation-of-state dependent commands are then included, and finally the definition is completed with command ENDEOS.

MAT *n type*
... material type-dependent commands, as required
ENDMAT
EOS *material_number type*
... equation-of-state dependent commands, as required
ENDEOS

TOPAZ material definitions begin with command TMAT followed immediately by the material heading and command MT specifying material type. Material type-dependent commands are then entered, as required. The command ENDMAT is NOT required to denoted the completion of a TOPAZ material definition.

TMAT *n*
heading
MT *type*
... material type-dependent commands, as required

ENDMAT	Terminate current material type definition.
MAT *n type*	Establish material number *n* of material type. Each material definition MUST be terminated by command ENDMAT.
MT *type* (TOPAZ)	Establish material type.
TMAT *n* *heading* (TOPAZ)	Establish material definition consisting of material number and a heading that is to be placed on the next input line.

Material Commands: DYNA2D

These commands modify DYNA2D materials globally.

DBQT *type*	Change default value of bulk viscosity type: EQ. 0: Default EQ. 1: Standard DYNA2D bulk viscosity EQ. 2: Richards-Wilkins bulk viscosity
DHGQ Q_h	Change default value of hourglass viscosity Q_h
DHQT *type*	Change default value of hourglass stabilization method: EQ. 0: Default EQ. 1: Standard DYNA2D EQ. 2: Rotational EQ. 3: Flanagan-Belytschko EQ. 4: Hancock EQ. 5: stiffness
DQL Q_l	Change default value of linear bulk viscosity $Q_{l.}$
DQQ Q_q	Change default value of quadratic bulk viscosity $Q_{q.}$

3.10.5 Equation-of-State Commands: DYNA2D

If a DYNA2D material requires an equation-of-state, the material definition (see above) must immediately be followed by command EOS. All relevant equation-of-state dependent commands are then included, and finally the definition is completed with command ENDEOS.

ENDEOS	Terminate equation-of-state specification.
EOS *material_number type*	Define equation-of-state *type* for *material_number.* Each equation-of-state definition is terminated by command ENDEOS.
HEAD *heading*	Equation-of-state *heading*. Heading must be entered in form: HEAD *heading*

3.10.6 Material Commands - Verbatim Mode

The material property “VERBATIM MODE” allows the declaration of material property conditions to be written exactly as specified by the user, subject to the keyword “BLANK” restriction (see below), in the MAZE output file.

The format of “VERBATIM MODE” within NIKE2D and DYNA2D is:

mat material_number 2000
. . . material properties to be copied in “verbatim mode”. . .
endmat

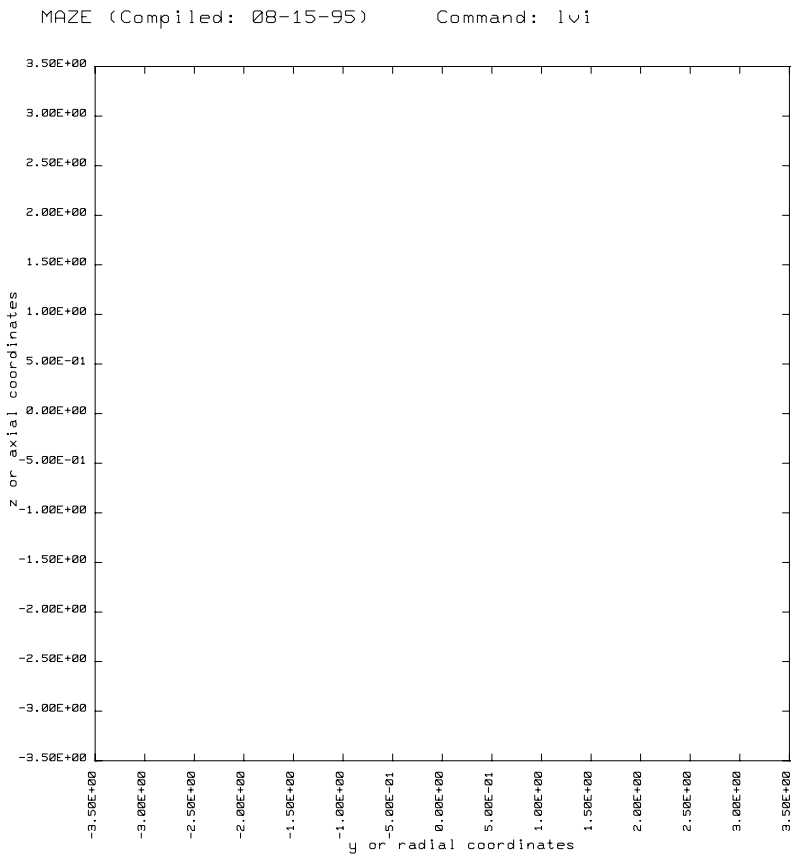
For “VERBATIM MODE” use in TOPAZ2D, a “dummy title” is required on the input line following the TOPAZ2D command “tmat” in accordance with TOPAZ2D material input format requirements. The “dummy title” will not appear in the MAZE output file. The format of “verbatim mode” for TOPAZ2D is:

tmat material_number 2000
“dummy title”
mt 2000
. . . material properties to be copied in “verbatim mode” . . .
endmat

The “VERBATIM MODE” keyword “BLANK” must be used to enter blank lines into the material property specifications of the model. No information following the keyword “BLANK” will be transferred to the MAZE output file until a carriage return is entered.

4 MAZE GEOMETRY AND PART DEFINITION

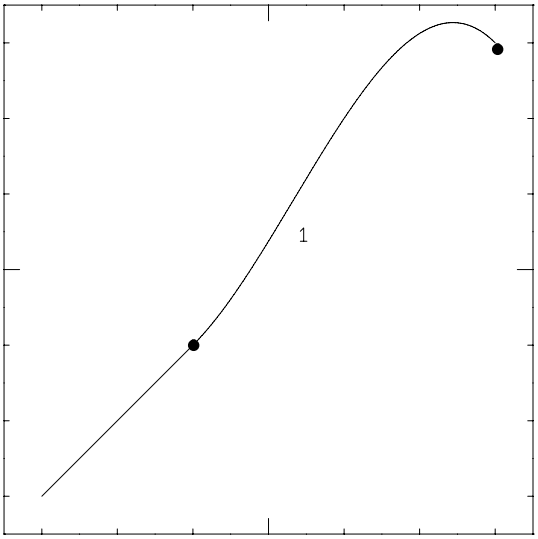
The following two chapter sections contain illustrations of the MAZE line and part definition commands. Common abscissa and ordinate ranges have been used for each diagram. These ranges were created using MAZE command: GSET 0 0 7. Axes labels and other identification have been omitted for purposes of page design format and space economy. The following diagram, however, presents a full illustration of the axes format in a MAZE line and part definition display.



CUBIC Θ_1 r_2 z_2 Θ_2

Define a free form line segment beginning at the most recently defined end point coordinates (r_1,z_1) with a slope of angle Θ_1 and ending at coordinates (r_2,z_2) with a slope of angle Θ_2. The line is defined using a cubic equation with four constants.

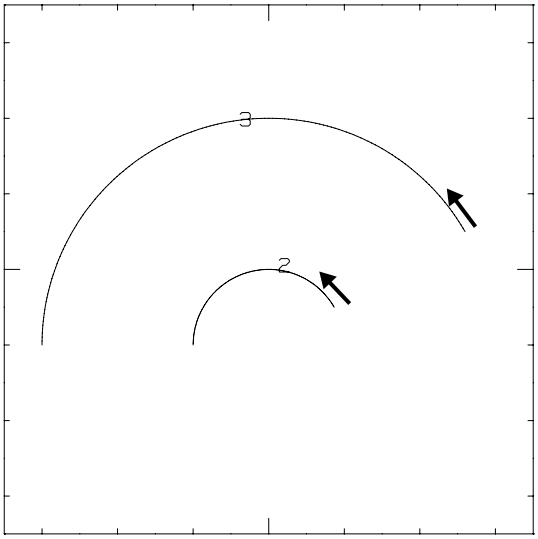
ld 1 lp 2 -3 -3 -1 -1
cubic 45.0 3.0 3.0 -45.0



LCC n r_c z_c Θ_1 Θ_2 r_1 ... r_n

Define n lines consisting of circular arcs centered at point (r_c,z_c) that sweep from angle Θ_1 to Θ_2 (degrees). The radii of the n lines are represented by r_1 ... r_n. MAZE will assign line numbers to the specified n lines to avoid conflicts with previously defined lines.

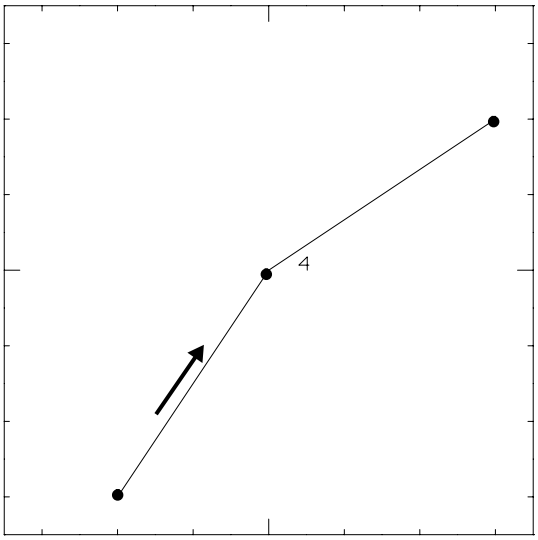
lcc 2 0 -1 30.0 180.0 1 3



LD m
LP n r_1 z_1 ... r_n z_n

Begin definition of line m.
Define n points (r_1,z_1) ... (r_n,z_n) to be added to the current line definition.

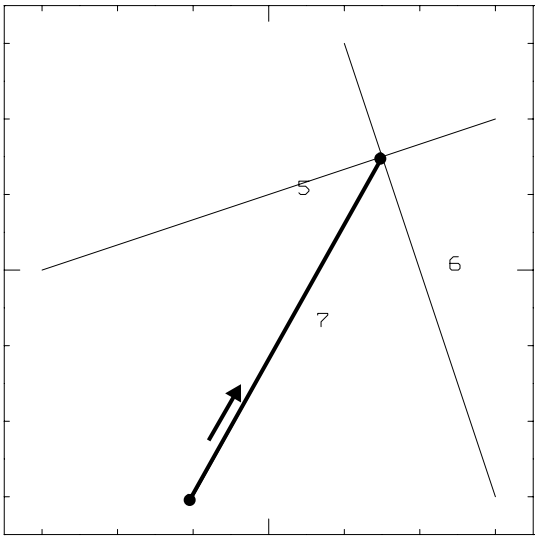
ld 4
lp 3 -2 -3 0 0 3 2



LPIL l_1 l_2

Define a point for the current line at the intersection of lines l_1 and l_2.

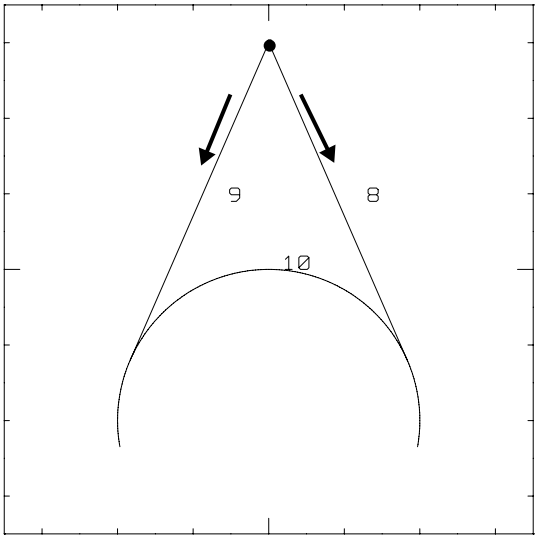
```
ld 5 lp 2 -3 0 3 2
ld 6 lp 2 3 -3 1 3
ld 7 lp 1 -1 -3 lpil 5 6
```



LPTA r_c z_c R

Define a line segment beginning at the last point defined and terminating at its tangency point on an arc of radius R, centered at point (r_c,z_c).

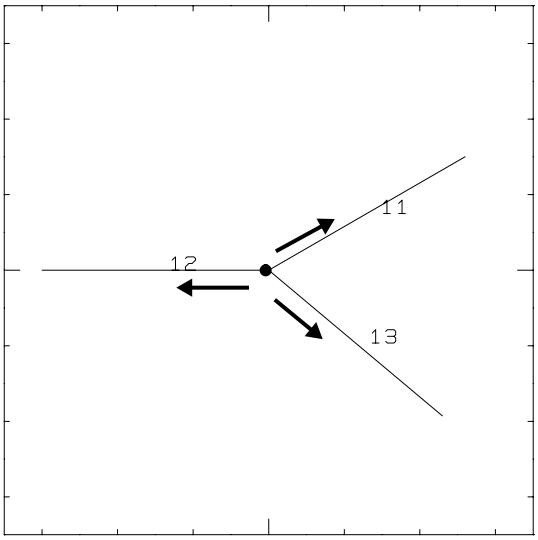
```
ld 8 lp 1 0 3
lpta 0 -2 2
ld 9 lp 1 0 3
lpta 0 -2 -2
lcc 1 0.0 -2 -10.0 190.0 2.0
```



LRL n r_c z_c l Θ_1 ... Θ_n

Define n radial lines of length l originating at point (r_c,z_c) and oriented at angles Θ_1 ... Θ_n (degrees).

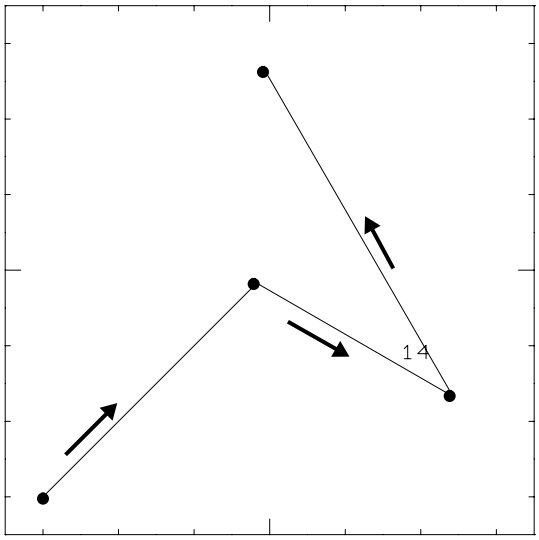
```
lrl 3 0 0 3 30 180 -40
```



LVC Θ *l*
LVC r_1 z_1 Θ *l*
LVC r_2 z_2 Θ *-l*

Define a line segment by a vector of length *l* oriented at Θ degrees. The vector begins at the last point defined (first command form) or at point (r_1,z_1) (second command form).

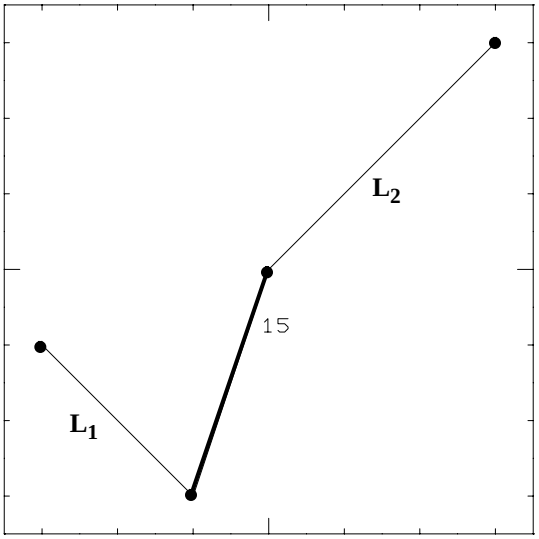
ld 14
lvc -3 -3 45 4
lvc -30 3
lvc 120 5



ML l_1 l_2

Append line l_2 to l_1.

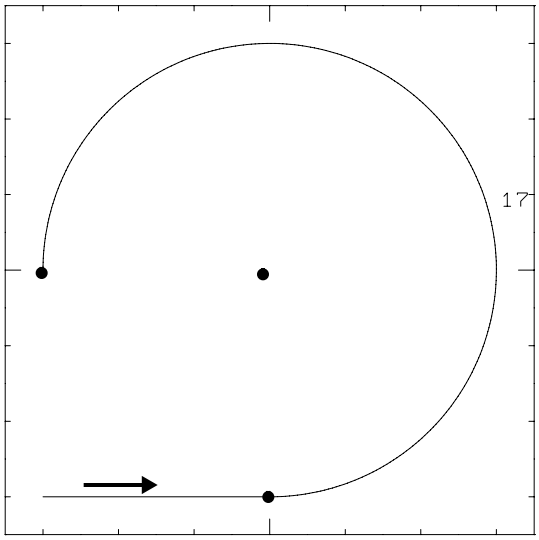
ld 15 lp 2 -3 -1 -1 -3
ld 16 lp 2 0 0 3 3
ml 15 16



CLAP r_1 z_1 r_c z_c

Define a circular arc centered at point (r_c,z_c) beginning at the last point defined and ending at point (r_1,z_1).

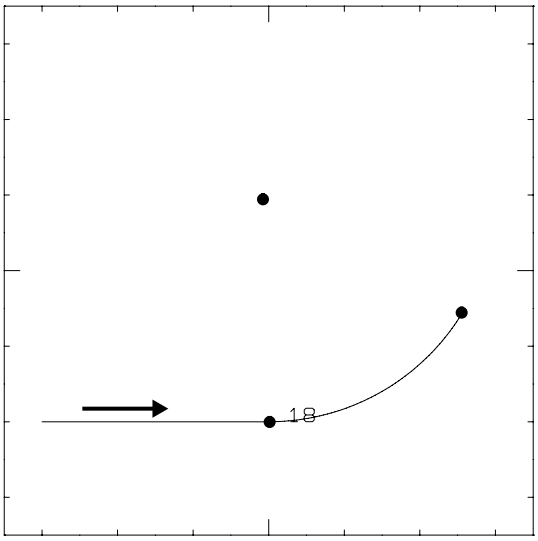
ld 17 lp 2 -3 -3 0 -3
clap -3 0 0 0



LAD r_c z_c Θ

Define a circular arc centered at point (r_c,z_c) beginning at the last point defined and sweeping through Θ degrees.

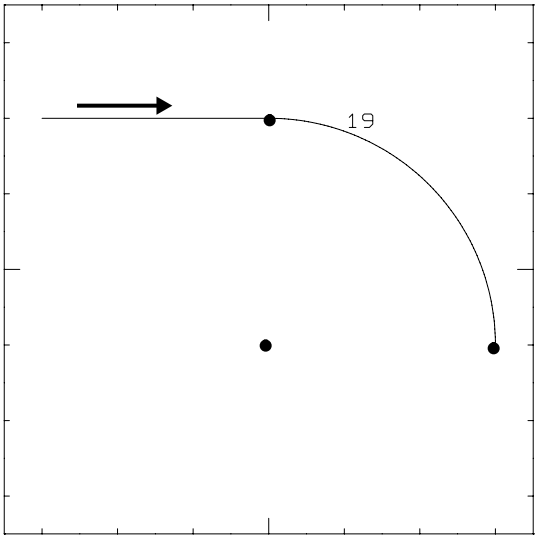
ld 18 lp 2 -3 -2 0 -2
lad 0 1 60



LAP r_1 z_1 r_c z_c

Define a circular arc centered at point (r_c,z_c) beginning at the last point defined and ending at point (r_1,z_1).

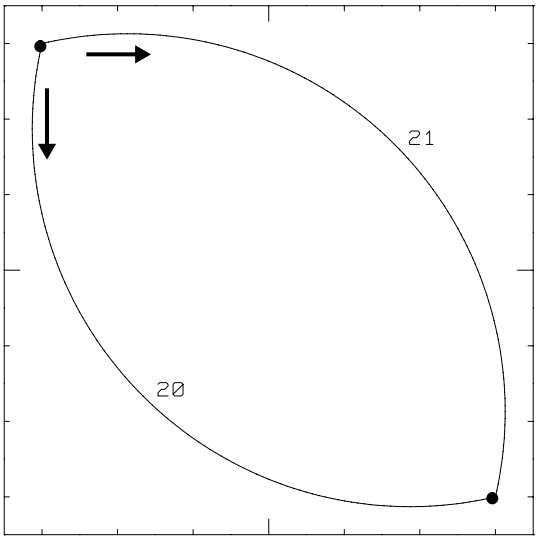
ld 19 lp 2 -3 2 0 2
lap 4 -1 0 -1
c NOTE: Arc need not intersect at (r_1,z_1).



LAR *r z R*

Define a circular arc of radius |*R*| beginning at the last point defined and ending at point (*r*,*z*).

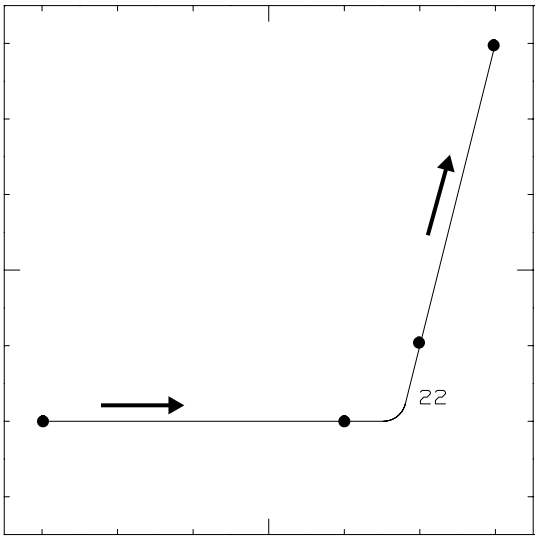
ld 20 lp 1 -3 3
lar 3 -3 5
ld 21 lp 1 -3 3
lar 3 -3 -5
c Centerpoint to left of arc 20.
c Centerpoint to right of arc 21.



LAT r_1 z_1 r_2 z_2 R

Define a circular arc $\leq 180^\circ$ of radius R tangent to the last line segment defined and tangent to the line segment joining point (r_1,z_1) to point (r_2,z_2).

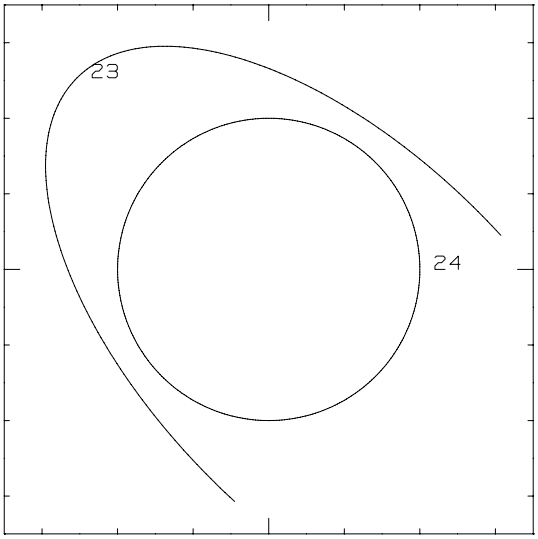
ld 22 lp 2 -3 -2 1 -2
lat 2 -1 3 3 0.3
c Note segment extensions to the arc tangency
c points.



LEP a b r_c z_c Θ_1 Θ_2 Φ

Define an elliptic arc centered at point (r_c,z_c) with semi-major axis a and semi-minor axis b. Arc sweeps from Θ_1 to Θ_2, as measured from the semi-major axis. Φ represents the inclination oft the semi-major axis from the positive r-axis.

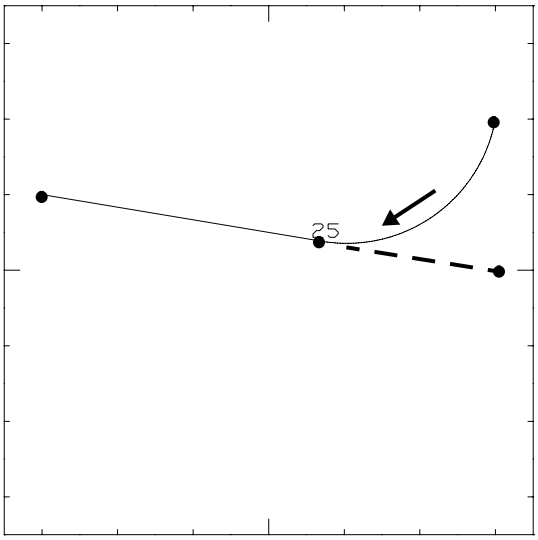
ld 23
lep 2.5 5 1 -1 -10 190 45
ld 24
lep 2 2 0 0 0 360 0



LPT r_1 z_1 r_2 z_2 R

Define a circular arc $\leq 180^\circ$ of radius R beginning at the last point defined and tangent to a line segment joining point (r_1,z_1) to point (r_2,z_2).

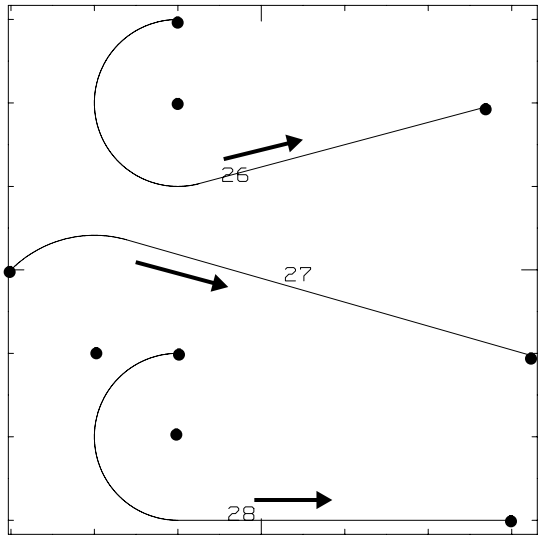
ld 25 lp 1 3 2
lpt 3 0 -3 1 2



LTAS r_{c1} z_{c1} *rot* r_{c2} z_{c2} R_2

Define a line segment of a circular arc centered around point (r_{c1},z_{c1}) followed by a straight line segment to a second tangency point.

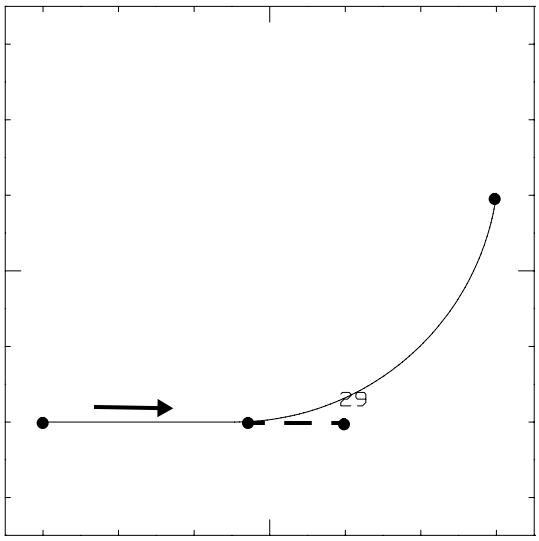
ld 26 lp 1 -1 3
ltas -1 2 1 3 1 1
ld 27 lp 1 -3 0
ltas -2 -1 -1 3 -2 1
ld 28 lp 1 -1 -1
ltas -1 -2 1 3 -2 -1



LTP *r z R*

Define a circular arc $\leq 180°$ of radius *R* tangent to the last line segment defined and terminating at point (*r*,*z*).

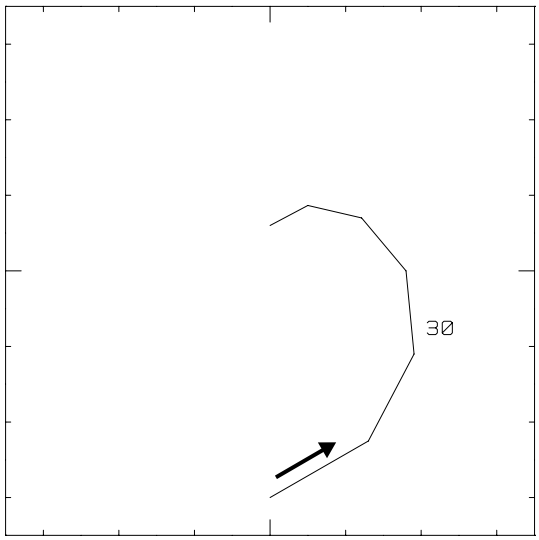
ld 29 lp 2 -3 -2 1 -2
ltp 3 1 3.5



LTBC *n* Θ ΔΘ *S* R_1 ... R_n

Define a line segment with tab cell data. Tab cell data consists of *n* radii, each separated by ΔΘ degrees, starting at angle Θ. Each radius is scaled by *S*.

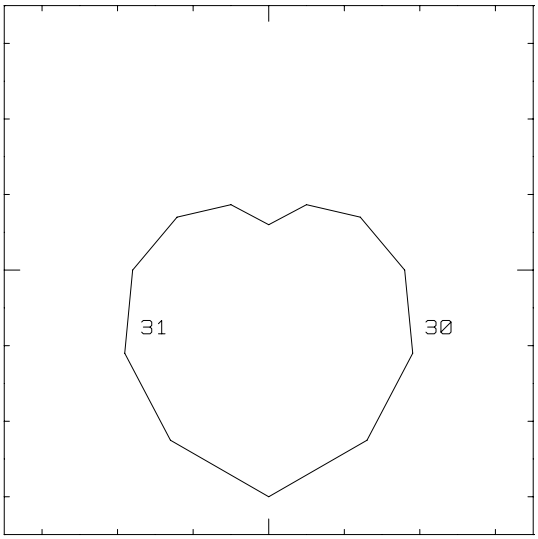
ld 30 ltbc 7 -90 30 1 3.0 2.6 2.2 1.8 1.4 1.0 0.6



LTBO m_1 δ_1 ... m_k δ_k

Define a line segment by offsetting the last line segment defined with the commands LTBC or LTBO. δ_1 is added to the radii of the first m_1 points, et cetera.

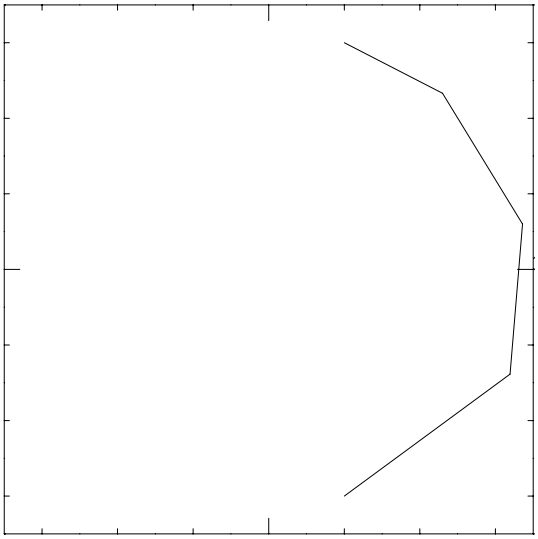
```
ld 31
ltbo 3 -3.6 4 -3.6
```



LO l r_1 z_1 r_2 z_2

Define a line segment by offsetting a segment of line l such that the new segment begins at point (r_1,z_1) and ends at point (r_2,z_2).

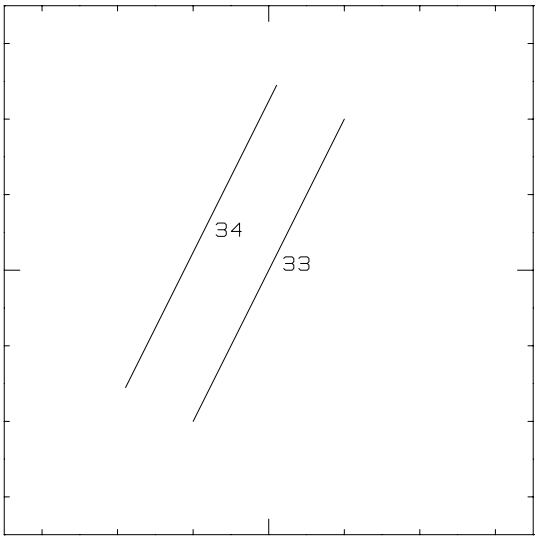
```
ld 32
lo 30 1 -3 1 3
c NOTE: Line 30 is defined with command
c “LTBC” (see above)
```



LOD l δ

Define a line segment by offsetting the entire line l a distance δ.

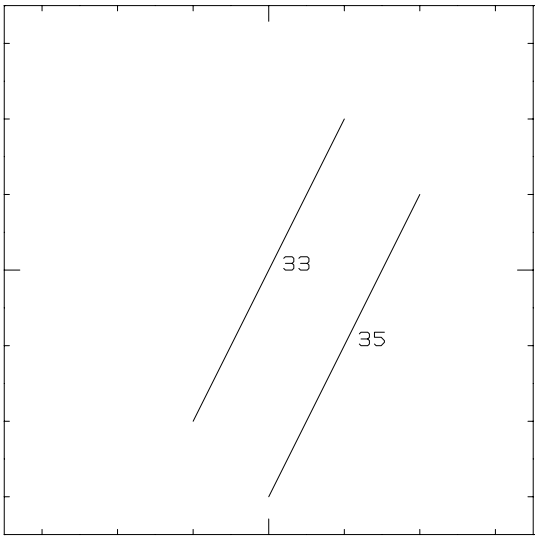
```
ld 33 lp 2 -1 -2 1 2
ld 34
lod 33 1
```



LSTL *l* Δr Δz

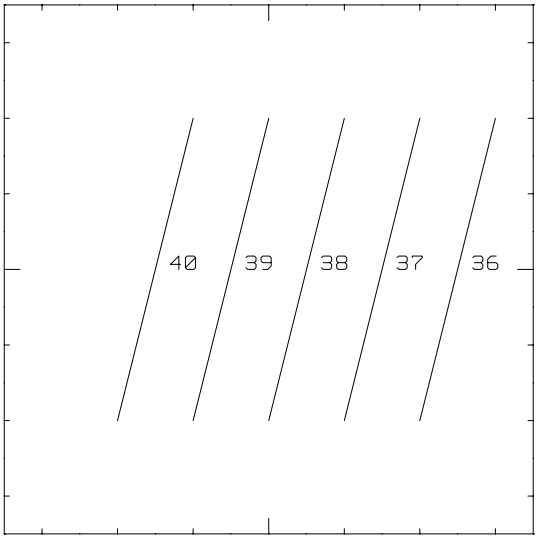
Define a line segment by translating the entire line *l* an offset of Δr and Δz.

```
ld 33 lp 2 -1 -2 1 2
ld 35
lstl 33 1 -1
```



LT *l* Δr Δz
LTM *n* l_1 ... l_n Δr Δz
LTS l_a ... l_b Δr Δz

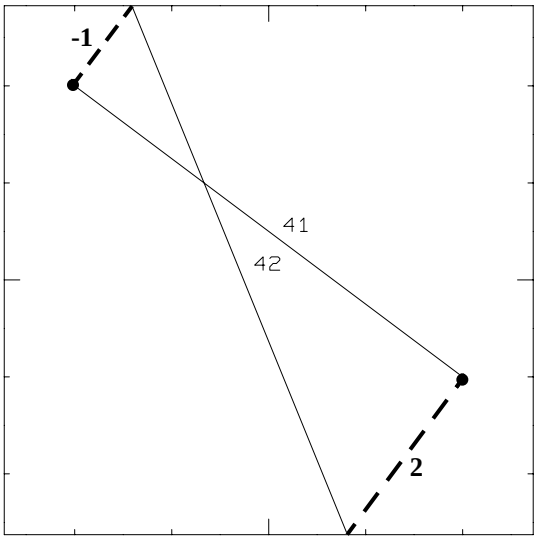
```
c Translate 1 line
ld 36 lp 2 1 -3 2 1
lt 36 1 1
c Translate 2 lines
ld 37 lp 2 0 -3 1 1
ld 38 lp 2 -1 -3 0 1
ltm 2 37 38 1.0 1.0
c Translate 2 additional lines
ld 39 lp 2 -2 -3 -1 1
ld 40 lp 2 -3 -3 -2 1
lts 39 40 1 1
```



VLOD *l* δ_1 δ_2

Define a line segment by offsetting the entire line *l* a distance δ_1 from the first point and a distance δ_2 from the last point. Intermediate points are linearly interpolated between δ_1 and δ_2.

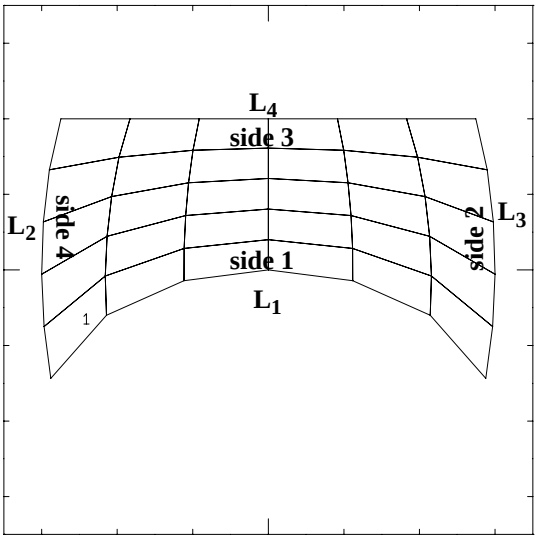
```
ld 41 lp 2 2 -1 -2 2
ld 42 vlod 41 2 -1
```



PART L_1 L_2 L_3 L_4 *mat k m*

Define the four sided region of material *mat* bounded by lines L_1, L_2, L_3, and L_4 to be a part with *k* elements along sides L_1 and L_3, *m* elements along sides L_2 and L_4.

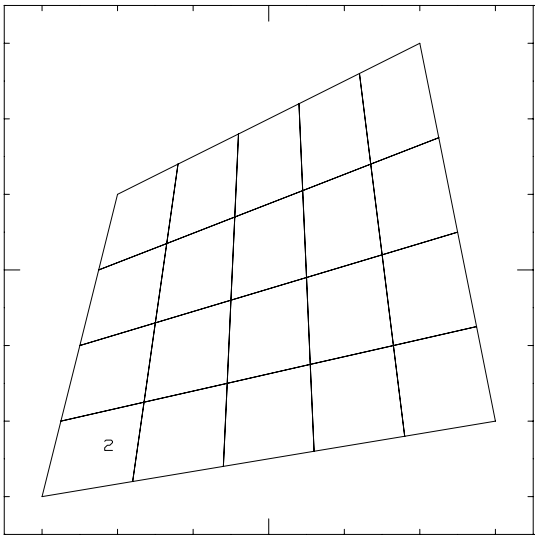
ld 1 lep 3 2 0 -2 180 0 0
ld 2 lep 3 1 -2 0 180 0 90
ld 3 lep 3 1 2 0 180 0 -90
ld 4 lp 2 3 2 -3 2
part 1 3 4 2 1 6 5 yes



QUAD r_1 z_1 r_2 z_2 r_3 z_3 r_4 z_4 *mat k m*

Define the four sided region of material *mat* bounded by corners r_1,z_1 ... r_4,z_4 to be a part with *k* elements along sides $r_1,z_1 \rightarrow r_2,z_2$ and $r_4,z_4 \rightarrow r_3,z_3$, *m* elements along sides $r_1,z_1 \rightarrow r_4,z_4$ and $r_2,z_2 \rightarrow r_3,z_3$.

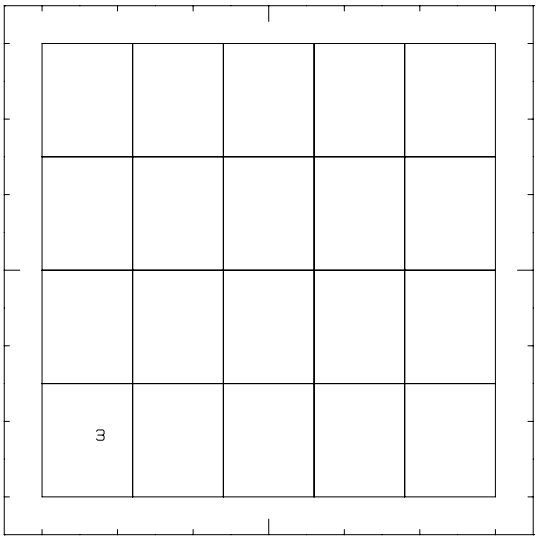
quad -3 -3 3 -2 2 3 -2 1 1 5 4 yes



RECT r_1 z_1 r_3 z_3 *mat k m*

Define the rectangular region of material *mat* bounded by opposite corners r_1,z_1 and r_3,z_3 to be a part with *k* elements in the *r* direction, *m* elements in the *z* direction.

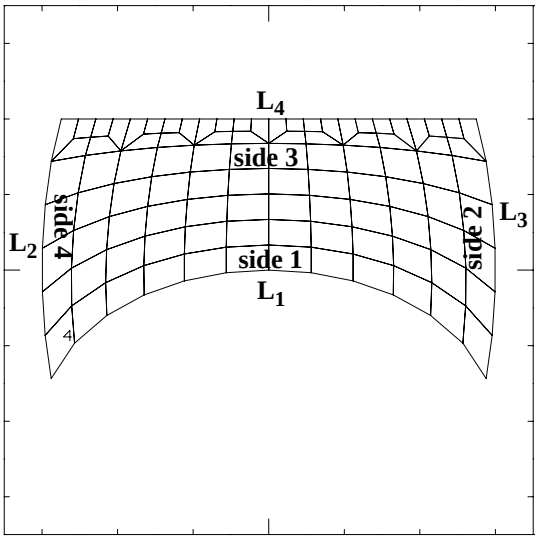
rect -3 -3 3 3 1 5 4 yes



T12

This command, typed as a preface to a PART, QUAD, or RECT command, causes the row of elements along part side L_3 to be subdivided into two times the number of elements in the other rows "parallel" to side L_1 and side L_3. If L_3 contains an odd number of elements, then one additional element is created on side L_2.

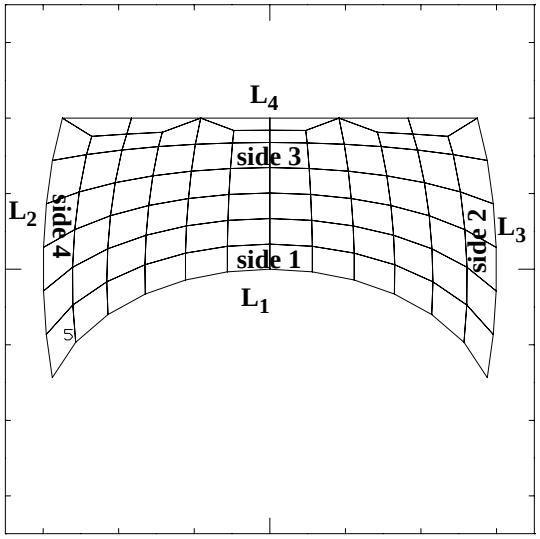
t12
part 1 3 4 2 1 12 6 yes



T21

This command, typed as a preface to a PART, QUAD, or RECT command, causes the row of elements along part side L_3 to be subdivided into one-half the number of elements in the other rows "parallel" to side L_1 and side L_3. Side L_3 must contain an even number of elements

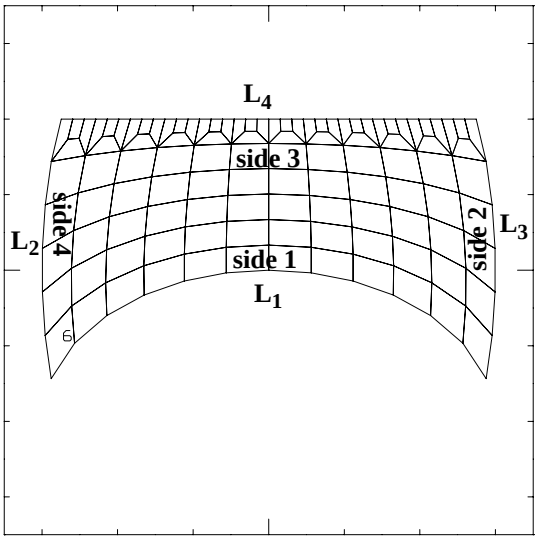
t21
part 1 3 4 2 1 12 6 yes



T13

This command, typed as a preface to a PART, QUAD, or RECT command, causes the row of elements along part side L_3 to be subdivided into three times the number of elements in the other rows "parallel" to side L_1 and side L_3.

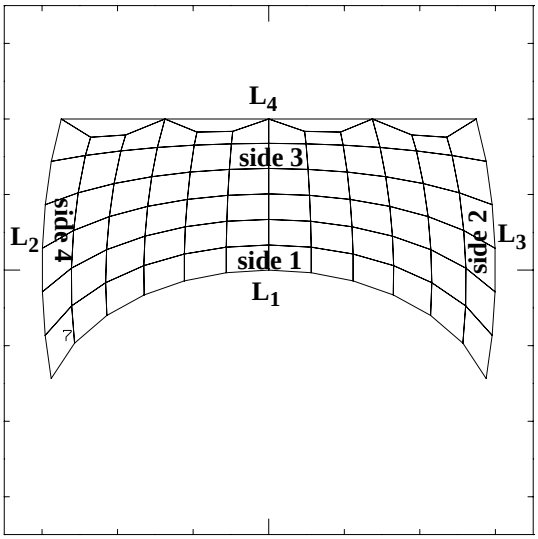
t13
part 1 3 4 2 1 12 6 yes



T31

This command, typed as a preface to a PART, QUAD, or RECT command, causes the row of elements along part side L_3 to be subdivided into one-third times the number of elements in the other rows "parallel" to side L_1 and side L_3. Side L_3 must contain a number of elements evenly divisible by 3.

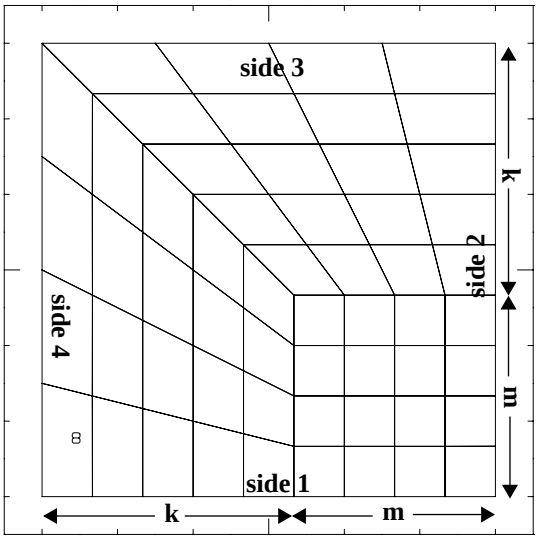
t13
part 1 3 4 2 1 12 6 yes



TRANS

This command, typed as a preface to a PART, QUAD, or RECT command, changes the part to one containing $k + m$ elements along sides L_1 and L_2 and m elements along sides L_3 and L_4.

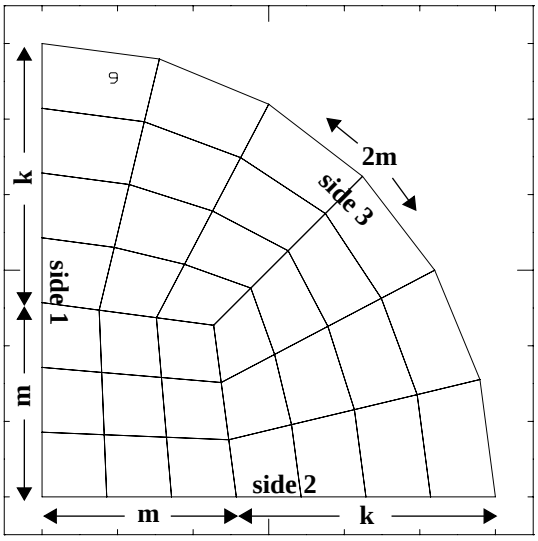
trans
rect -3 -3 3 3 1 5 4 yes



PART *L_1 L_2 L_3 L_3 mat k m*

Define the three sided region of material *mat* bounded by lines L_1, L_2, and L_3 to be a part with $k + m$ elements along sides L_1 and L_2, $2m$ elements along side L_3.

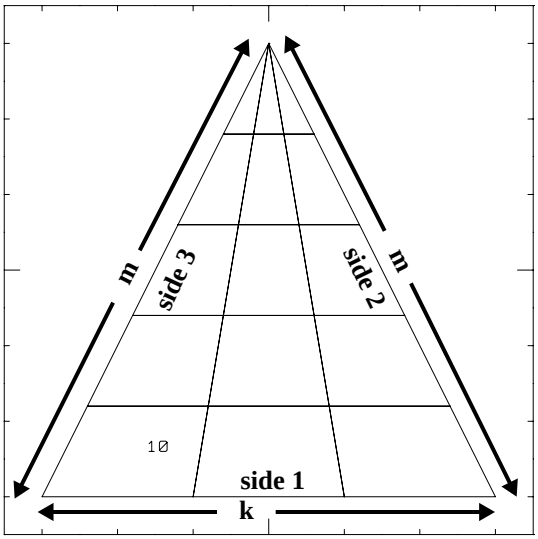
ld 5 lp 2 -3 -3 -3 3
ld 6 lp 2 -3 -3 3 -3
lcc 1 -3 -3 0 90 6
part 5 6 7 7 1 4 3 yes



PART L_1 L_2 L_3 *0 mat k m*

Define the three sided region bounded by lines L_1, L_2, and L_3 to be a part with k elements along side L_1, m elements along sides L_2 and L_3, and material *mat*. k triangular elements will exist at the intersection of lines L_2 and L_3.

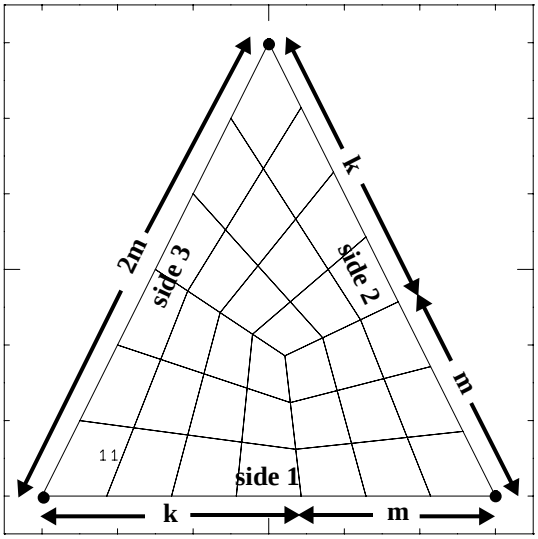
ld 8 lp 2 3 -3 0 3
ld 9 lp 2 0 3 -3 -3
part 6 8 9 0 1 3 5 yes
c NOTE: Line 6 previously defined (see above).



TRIQ r_1 z_1 r_2 z_2 r_3 z_3 *mat k m*

Define the three sided region bounded by points (r_1,z_1), (r_2,z_2), and (r_3,z_3) to be a part with $k + m$ elements along side one $(r_1,z_1 \rightarrow r_2,z_2)$ and side two $(r_2,z_2 \rightarrow r_3,z_3)$, $2m$ elements along side three $(r_3,z_3 \rightarrow r_1,z_1)$, and material *mat.*

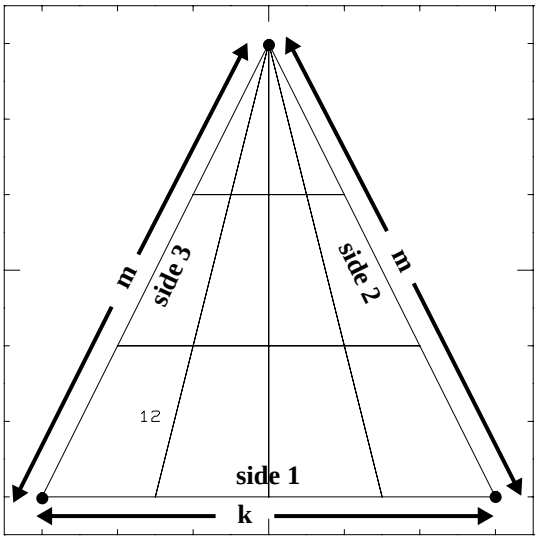
triq -3 -3 3 -3 0 3 1 4 3 yes



TRIT r_1 z_1 r_2 z_2 r_3 z_3 *mat k m*

Define the three sided region bounded by points (r_1,z_1), (r_2,z_2), and (r_3,z_3) to be a part with k elements along side one $(r_1,z_1 \rightarrow r_2,z_2)$, m elements along side two $(r_2,z_2 \rightarrow r_3,z_3)$ and side three $(r_3,z_3 \rightarrow r_1,z_1)$, and material *mat.*

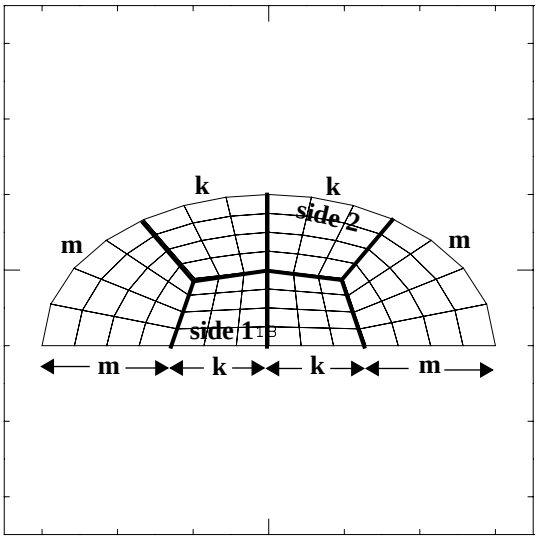
trit -3 -3 3 -3 0 3 1 4 3 yes



PART L_1 L_2 L_2 L_2 *mat k m*

Define the region of material *mat* bounded by lines L_1 and L_2 (one line must be straight and one line must be an arc) to be a part with $m + k + k + m$ elements ($k \geq 3$; $m \geq 2$) along sides L_1 and L_2.

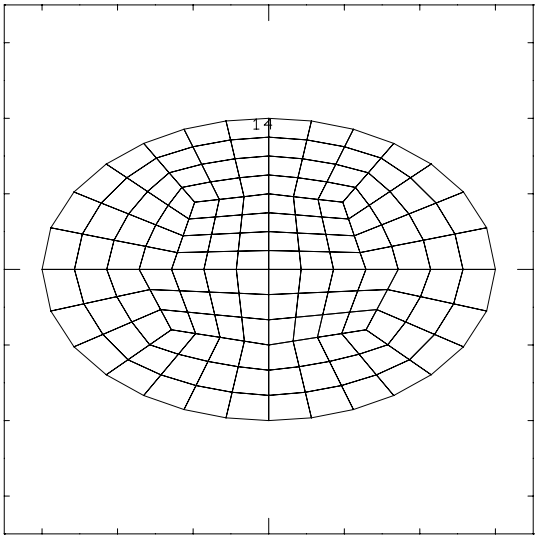
```
ld 10 lp 2 -3 -1 3 -1
ld 11 lep 3 2 0 -1 -10 190 0
part 10 11 11 11 1 3 4 yes
```



PART L_1 L_1 L_1 L_1 *mat k m*

Define the region of material *mat* bounded by elliptic arc L_1 to be a part with $2(m + k + k + m)$ elements ($k \geq 3$; $m \geq 3$) along arc L_1.

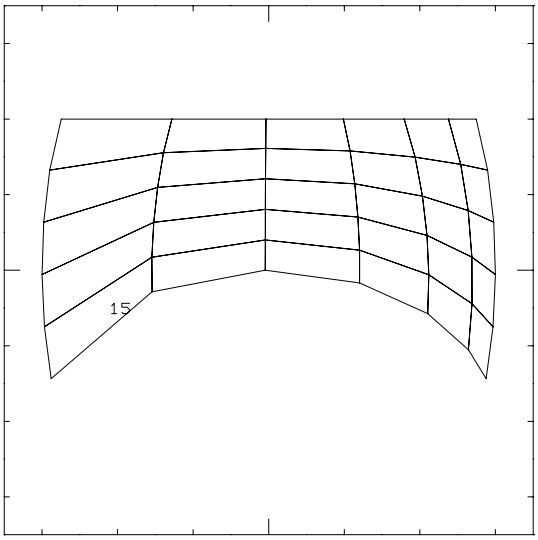
```
ld 12 lep 3 2 0 0 0 360 0
part 12 12 12 12 1 3 4 yes
```



PART ... }
QUAD ... } *mat -k m R1*
RECT ... }

Define a four sided part of material *mat* with *R1* nodal spacing and element transitions along edges L_1 and L_3.

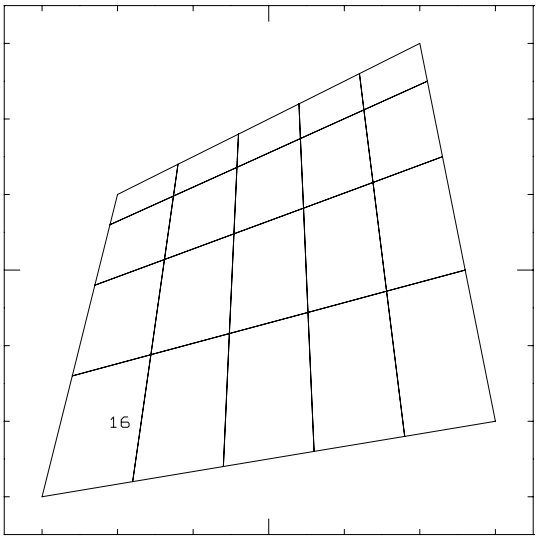
```
part 1 3 4 2 1 -6 5 4 yes
```



PART ...
QUAD ... } *mat k -m R2*
RECT ...

Define a four sided part of material *mat* with *R2* nodal spacing and element transitions along edges L_2 and L_4.

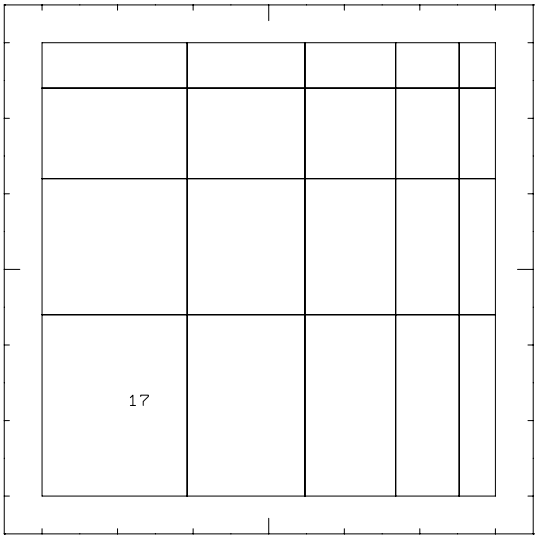
quad -3 -3 3 -2 2 3 -2 1 1 5 -4 4 yes



PART ...
QUAD ... } *mat -k -m R1 R2*
RECT ...

Define a four sided part of material *mat* with *R1* nodal spacing and element transitions along edges L_1 and L_3 and *R2* nodal spacing and element transitions along edges L_2 and L_4.

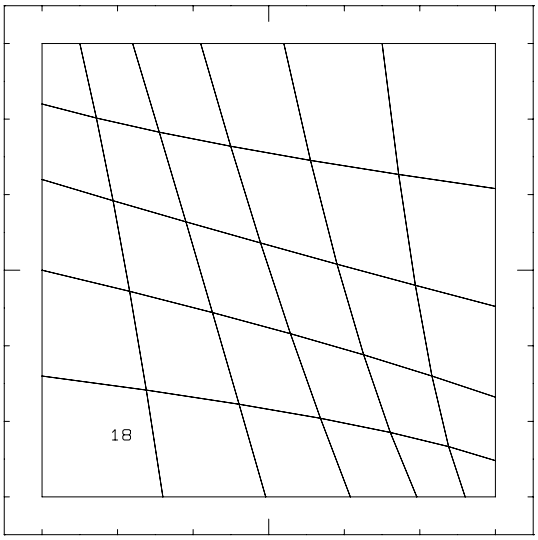
rect -3 -3 3 3 1 -5 -4 4 4 yes



PART ...
QUAD ... } *-mat k m* R_1 R_2 R_3 R_4
RECT ...

Define a four sided part with independent nodal spacing and element sizing that transitions smoothly across the part.

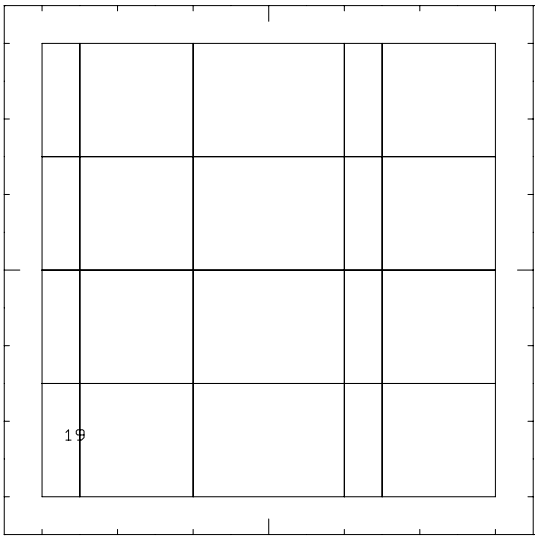
rect -3 -3 3 3 -1 6 5 4 0.25 3 0.5 yes



PART L_1 L_2 L_3 L_4 *mat 0 m*

Construct a four sided part of material *mat* in which the line definition points on edges L_1 and L_3 define the element transitions.

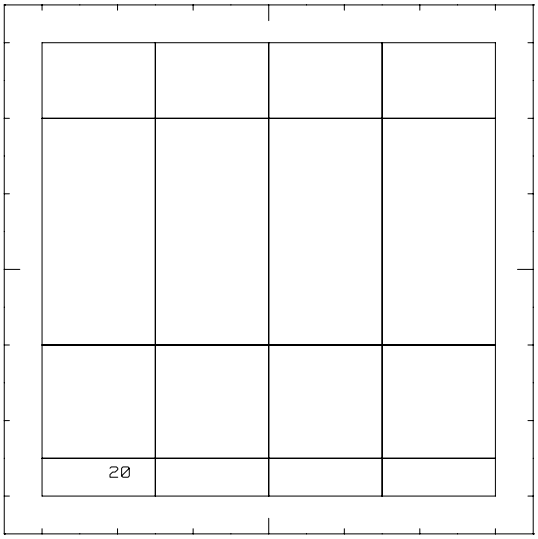
```
ld 13 lp 6 -3 -3 -2.5 -3 -1 -3 1 -3 1.5 -3 3 -3
ld 14 lp 5 3 -3 3 -2.5 3 -1 3 2 3 3
ld 15 lstl 13 0 6
ld 16 lstl 14 -6 0
part 13 14 15 16 1 0 4 yes
```



PART L_1 L_2 L_3 L_4 *mat k 0*

Construct a four sided part of material *mat* in which the line definition points on edges L_2 and L_4 define the element transitions.

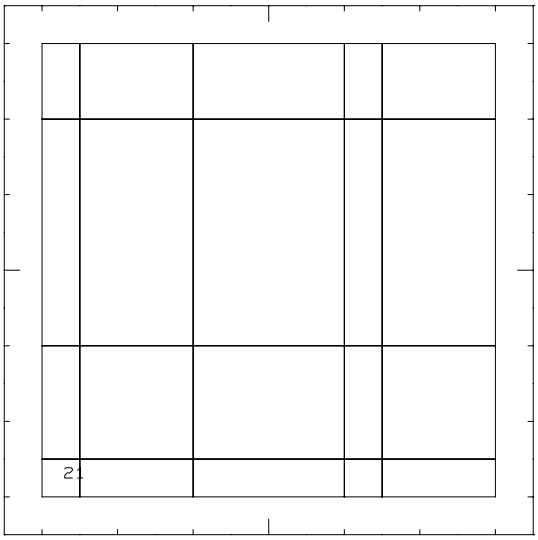
```
part 13 14 15 16 1 4 0 yes
```



PART L_1 L_2 L_3 L_4 *mat 0 0*

Construct a four sided part of material *mat* in which all the line definition points define the element transitions.

```
part 13 14 15 16 1 0 0 yes
```



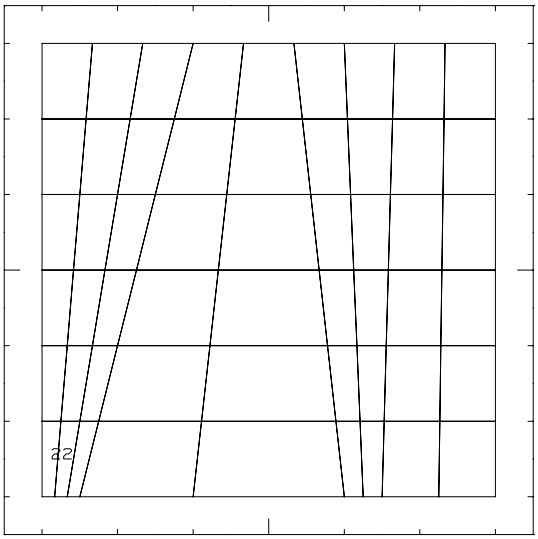
PART -L_a L_b L_c L_d *mat k m* $n^a{}_1$ … $n^a{}_{pa-2}$

...

et cetera

Construct a four sided part of material *mat* in which the points in line definition L_a define the element transitions along that edge.

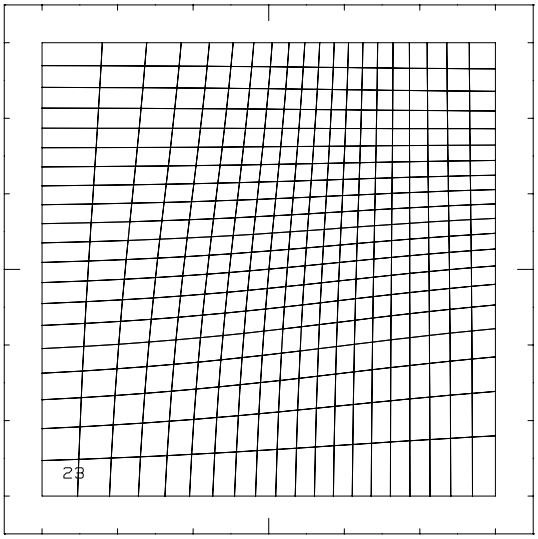
part -13 14 15 16 1 9 6 3 1 1 2 yes



AZON *n* S_1 … S_n r_c z_c

In subsequent PART commands, nodes will be distributed along sides $S_{i;\,(1 \le i \le 4)}$ using equal angular spacing based on the center point (r_c, z_c).

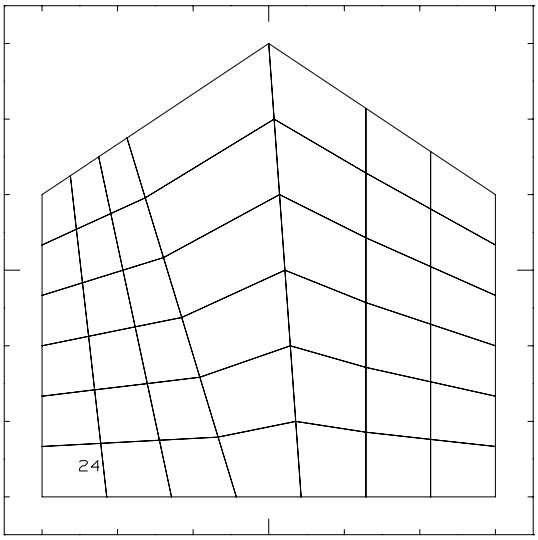
azon 4 1 2 3 4 1 1
part 13 14 15 16 1 20 20 yes
azoff
c NOTE: Angular positioning should be turned
c off after use.



AOR Θ

MAZE will attempt to establish a node at the vertex of "sharp" angles < Θ in part boundary lines in order that the angles be preserved within the part description.

ld 17 lp 2 -3 -3 3 -3
ld 18 lp 2 3 -3 3 1
ld 19 lp 3 -3 1 0 3 3 1
ld 20 lp 2 -3 -3 -3 1
aor 115
part 17 18 19 20 1 7 6 yes



5 DYNA2D MATERIAL PROPERTY COMMANDS

This chapter contains MAZE commands used to specify DYNA2D material model parameters and a brief theoretical explanation of the particular model. The user is referred to "DYNA2D: A Nonlinear, Explicit, Two-Dimensional Finite Element Code for Solid Mechanics - User Manual" to obtain all references mentioned.

5.1 General Material Definition Commands

The following commands apply to the current material only, and may override previous default values.

head *heading*	Material identification. Command is entered in the form: head *heading*.
ro ρ	Material density
hgqt	Hourglass stabilization method: EQ. 0: Default EQ. 1: Standard DYNA2D EQ. 2: Rotational EQ. 3: Flanagan-Belytschko EQ. 4: Hancock EQ. 5: stiffness
hgq Q_h	Hourglass viscosity coefficient (Default: $Q_h = 0.10$): EQ. 1: $Q_h \leq 0.15$ for stability EQ. 2: $Q_h \leq 0.20$ for stability EQ. 3: $Q_h \leq 0.40$ for stability EQ. 4: $Q_h \leq 0.40$ for stability
bqt	Bulk viscosity type: EQ. 0: Default EQ. 1: Standard DYNA2D bulk viscosity EQ. 2: Richards-Wilkins bulk viscosity
bqq Q_q	Quadratic shock viscosity coefficient (Default: $Q_q = 1.5$)
bql Q_l	Linear shock viscosity coefficient (Default: $Q_l = 0.06$)
srdr *rate*	Stress rate default reset: EQ. 0: DYNA2D default stress rate for this material EQ. 1: Jaumann rate EQ. 2: Green-Naghdi rate

5.2 DYNA2D Material Type 1: Elastic

Command	Variable	Description
e	E	Young's modulus
pr	v	Poisson's ratio

This model produces isotropic, linear elastic material behavior.

5.3 DYNA2D Material Type 2: Orthotropic Elasticity

Command	Variable	Description
ea	E_a	Elastic modulus in *a*-direction
eb	E_b	Elastic modulus in *b*-direction
ec	E_c	Elastic modulus in *c*-direction
prba	v_{ba}	Poisson's ratio, *ba*
prca	v_{ca}	Poisson's ratio, *ca*
prcb	v_{cb}	Poisson's ratio, *cb*
gab	G_{ab}	Shear modulus, *ab*
aopt	*option*	Material axes definition: EQ. 0.0: locally orthotropic with material axes determined by angle Ψ and nodes n_1 and n_2 specified on each element card EQ. 1.0: locally orthotropic with material axes determined by a point in space and the global location of Gauss integration points EQ. 2.0: globally orthotropic with material axes determined by angle Ψ_G.
yp or rp	y_p or r_p	Coordinate y_p or r_p (defined for aopt = 1.0)
zp	z_p	Coordinate z_p (defined for aopt = 1.0)
psig	Ψ_G	Angle Ψ_G (radians; defined for aopt = 2.0)

The constitutive matrix $\mathbf{C}$ relating increments in stress to increments in strain is defined as

$$\mathbf{C} = \bar{\mathbf{T}}^T \mathbf{C}_L \bar{\mathbf{T}}, \tag{4-1}$$

where $\bar{\mathbf{T}}$ is the appropriate transformation matrix and $\mathbf{C}_L$ is the constitutive matrix defined in terms of the orthogonal material axes, a and b,

$$\mathbf{C}_L^{-1} = \begin{bmatrix} \frac{1}{E_a} & -\frac{\nu_{ba}}{E_b} & -\frac{\nu_{ca}}{E_c} & 0 \\ -\frac{\nu_{ab}}{E_a} & \frac{1}{E_b} & -\frac{\nu_{cb}}{E_c} & 0 \\ -\frac{\nu_{ac}}{E_a} & -\frac{\nu_{bc}}{E_b} & \frac{1}{E_c} & 0 \\ 0 & 0 & 0 & \frac{1}{G_{ab}} \end{bmatrix}. \quad (4\text{-}2)$$

Poisson's ratios are defined as

$$\nu_{ij} = \frac{-\varepsilon_j}{\varepsilon_i} \quad (4\text{-}3)$$

which represents the strain ratio resulting from a uniaxial stress applied in the *i-th* direction.

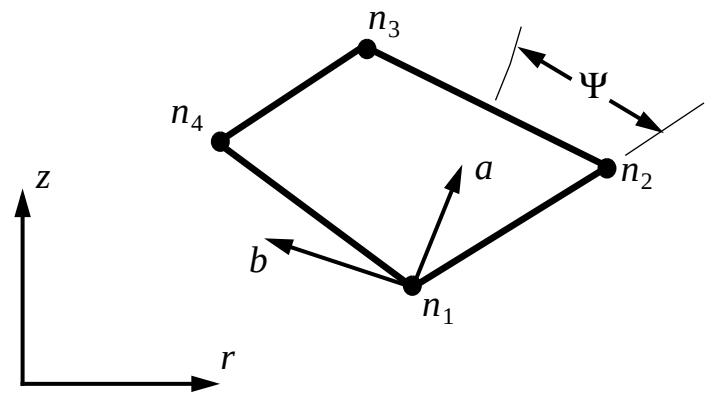
5.4 DYNA2D Material Type 3: Kinematic/Isotropic Elastic-Plastic

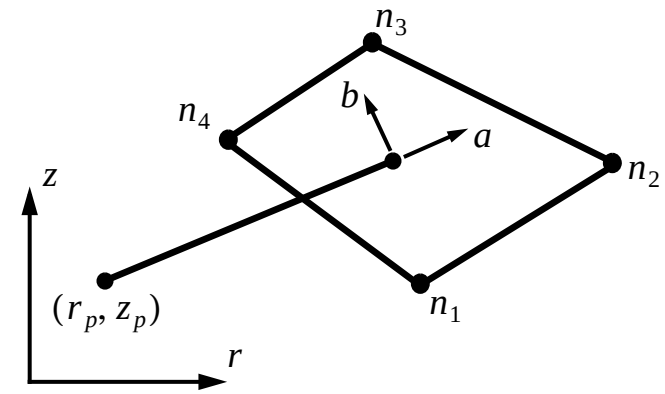
Command	Variable	Description
e	E	Young's modulus
v	ν	Poisson's ratio
sigy	σ_o	Yield stress
etan	E_T	Tangent modulus
beta	β	Hardening parameter
epsf	$\bar{\varepsilon}^p$	Effective plastic strain at failure

The material behavior is elastoplastic and includes linear strain hardening and material failure. The hardening parameter β specifies an arbitrary combination of kinematic and isotropic hardening; $\beta = 0.0$ yields purely kinematic hardening, while $\beta = 1.0$ gives purely isotropic hardening. Figure 4.2 illustrates the effect of β on the uniaxial stress-strain curve.

Material ‘‘erosion’’ and failure may be obtained by defining a nonzero effective plastic strain at failure $\bar{\varepsilon}_f^p$ and specifying that this material is active for automatic contact.

The yield condition can be written as





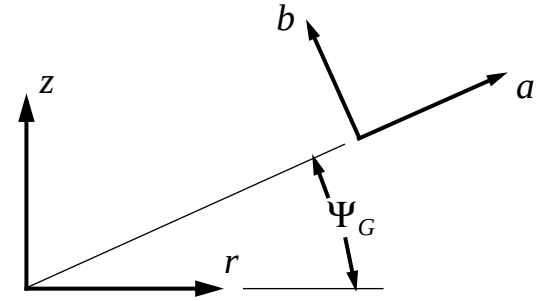


Figure 4.1
Options for determining the principal material axes
(a) AOPT = 0.0; (b) AOPT = 1.0; (c) AOPT = 2.0.

$$\phi = \bar{\sigma} - \sigma_y(\bar{\varepsilon}^p) , \tag{4-4}$$

where $\bar{\sigma}$ is the effective stress and σ_y is the current yield stress, which may be a function of the effective plastic strain $\bar{\varepsilon}^p$ if strain hardening is included. For isotropic hardening, the effective stress $\bar{\sigma}$ is given by

$$\bar{\sigma} = \left(\frac{3}{2}s_{ij}s_{ij}\right)^{\frac{1}{2}} , \tag{4-5}$$

where s_{ij} is the deviatoric stress tensor. For kinematic hardening,

$$\bar{\sigma} = \left(\frac{3}{2}\eta_{ij}\eta_{ij}\right)^{\frac{1}{2}} \tag{4-6}$$

where the translated stress η_{ij} is defined as

$$\eta_{ij} = s_{ij} - \alpha_{ij} , \tag{4-7}$$

and α_{ij} is the (deviatoric) back stress tensor.

The linear isotropic hardening law has the form

$$\sigma_y = \sigma_0 + \beta E_p \bar{\varepsilon}^p, \tag{4-8}$$

where σ_y is the current yield stress, σ_0 is the initial yield stress, and E_p is the plastic modulus.

The effective plastic strain $\bar{\varepsilon}^p$ is given by

$$\bar{\varepsilon}^p = \int_0^t d\bar{\varepsilon}^p , \tag{4-9}$$

where $d\bar{\varepsilon}^p$ is the incremental effective plastic strain. The plastic modulus is found from Young's modulus E and the tangent modulus E_T using

$$E_p = \frac{EE_T}{E - E_T} . \tag{4-10}$$

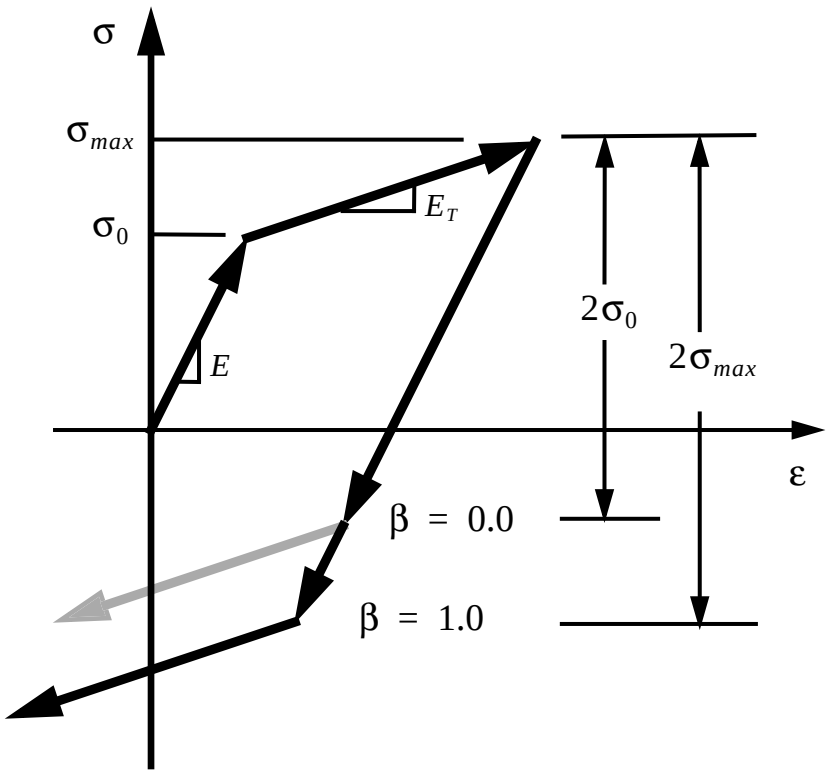


Figure 4.2
Uniaxial stress-strain curve showing elastic-plastic material behavior for kinematic hardening ($\beta = 0$) and isotropic hardening ($\beta = 1.0$).

The plastic hardening modulus E_p is the slope of the inelastic portion of the effective stress $\bar{\sigma}$ vs. effective plastic strain $\bar{\varepsilon}^p$ curve. Similarly, the tangent modulus E_T is the slope of the inelastic part of a uniaxial stress vs. strain curve (or equivalently, the effective stress vs. effective strain curve).

5.5 DYNA2D Material Type 4: Thermo-Elastic-Plastic

Command	Variable	Description
npts	n	Number of temperature values for defined material constants
temp	$T_1 \ldots T_n$	Temperatures
e	$E_1 \ldots E_n$	Young's modulus at T_i
pr	$\nu_1 \ldots \nu_n$	Poisson's ratio at T_i
alpha	$\alpha_1 \ldots \alpha_n$	Secant coefficients of thermal expansion
sigy	$\sigma_{y1} \ldots \sigma_{yn}$	Yield stress at T_i
etan	$E_{t1} \ldots E_{tn}$	Plastic modulus at T_i

At least two temperatures and their corresponding material properties must be defined. The analysis will be terminated if a material temperature falls outside the range defined in the input.

The plastic hardening modulus E_p is the slope of the effective stress vs. effective plastic strain curve (or equivalently, the uniaxial stress vs. effective plastic strain curve). The plastic hardening modulus may be found from the tangent modulus E_T as

$$E_p = \frac{E E_T}{E - E_T}, \tag{4-11}$$

where tangent modulus E_T is the slope of the post-yield portion of the uniaxial stress - strain curve.

Thermal expansion due to temperature change is included when nonzero values of $\bar{\alpha}$ are specified. The *secant* coefficient of thermal expansion $\bar{\alpha}$ can also be a function of temperature, and is defined with respect to the reference temperature at the beginning of the calculation for that material. Total thermal strain ε_{ij}^{T} is defined in terms of the secant thermal expansion coefficient $\bar{\alpha}$ as

$$\varepsilon_{ij}^{T} = \bar{\alpha}(T - T_{ref})\delta_{ij}\,, \tag{4-12}$$

where T is the current temperature and T_{ref} is the reference temperature. Therefore, temperature dependent secant coefficients of thermal expansion should be defined as the value *to* that temperature, not the value *at* that temperature. The secant coefficient $\bar{\alpha}$ is related to the tangent coefficient of thermal expansion α by

$$\bar{\alpha} = \frac{1}{T - T_{ref}} \int_{T_{ref}}^{T} \alpha(T)\,dT\,. \tag{4-13}$$

For temperature independent coefficients of thermal expansion, $\bar{\alpha}$ is identical to α, and the classical definition of thermal expansion is valid.

The reference temperature in this model is chosen as the first temperature in the TOPAZ2D plot files, or the temperature at time $t = 0.0$ if temperature is specified by a load curve.

5.6 DYNA2D Material Type 5: Soil and Crushable Foam

Command	Variable	Description
g	G	Shear modulus
ku	K_u	Bulk unloading modulus
a0	a_0	Yield function constant, a_0
a1	a_1	Yield function constant, a_1, or load curve, NC_1, giving pressure vs. volumetric strain
a2	a_2	Yield function constant, a_2, or load curve, NC_2, giving yield stress vs. pressure
pc	p_{cut}	Pressure cutoff
npts	n	Number of points in volumetric strain vs. pressure curve
vs	$\varepsilon_{v1} \ldots \varepsilon_{vn}$	Volumetric strain
p	$p_1 \ldots p_n$	Pressures
amod	*flag*	Modified elliptical surface flag: EQ. 0.0: unmodified surface EQ. 1.0: failure surface is constant at high pressures

This model has two options: one using an analytical function to describe the variation of yield stress with pressure, and one using a load curve to define a pressure-dependent yield stress. The analytical form is obtained if constant a_0 is input as a positive number (required for a physically meaningful analytical model), and the load curve form is obtained if a_0 is input as a negative number. If the load curve form is chosen, load curve NC_1 is used for to describe the pressure p vs. volumetric strain ε_v curve. Load curve NC_2 is then used to directly specify yield stress σ_y as a function of pressure.

Pressure is positive in compression, and volumetric strain is negative in compression. Volumetric strain is given by the natural logarithm of the relative volume. Note $\varepsilon_v = \ln\left(\frac{V}{V_o}\right)$. The tabulated pressure-volumetric strain data may contain up to nine pairs of points, and must be given in order of *increasing* compression. If the pressure drops below (i.e., becomes more tensile than) the cutoff value p_{cut}, then the pressure is reset to the cutoff value.

The deviatoric perfectly-plastic yield function ϕ is defined as

$$\phi = J_2 - [a_0 + a_1 p + a_2 p^2], \tag{4-14}$$

where a_0, a_1, and a_2 are constants, p is pressure, and J_2 is the second invariant of the deviatoric stress tensor $\boldsymbol{s}$ given by

$$J_2 = \frac{1}{2} s_{ij} s_{ij}. \tag{4-15}$$

Plastic flow is nonassociative if a_1 or a_2 are nonzero. On the yield surface, $J_2 = \frac{1}{3}\sigma_y^2$, where σ_y is the uniaxial yield stress. Thus, the yield stress at any pressure p is given by

$$\sigma_y = [3(a_0 + a_1 p + a_2 p^2)]^{\frac{1}{2}}. \tag{4-16}$$

There is no strain hardening in this model, so the yield stress is completely determined by the pressure.

To eliminate the pressure dependence of the yield strength, set $a_1 = a_2 = 0$ and $a_0 = \frac{1}{3}\sigma_y^2$. This approach is useful when a von Mises type elastic-plastic model is desired for use with tabulated volumetric data.

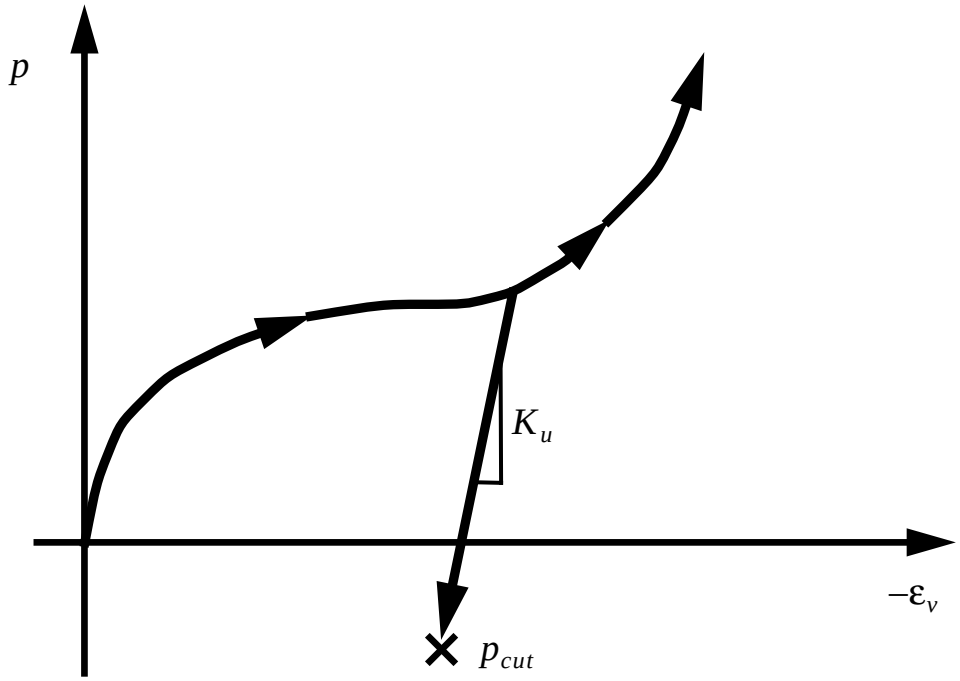


Figure 4.3
Pressure vs. volumetric strain curve for soil and crushable foam model.

The variation of σ_y as a function of p has three conical forms which depend on the parameter a_2: elliptic ($a_2 < 0$), parabolic ($a_2 = 0$), or hyperbolic ($a_2 > 0$). These three forms are shown in Figure 4.4 in terms of J_2 vs. p. Figure 4.5 shows the corresponding forms in a more familiar engineering form of σ_y vs. p. If the modified elliptical yield surface flag $AMOD$ is nonzero, then the elliptical yield surface is used up to the point of maximum J_2. For higher pressures, the yield surface is extended as a von Mises surface. The resulting yield surface is depicted in Figure 4.4 and Figure 4.5.

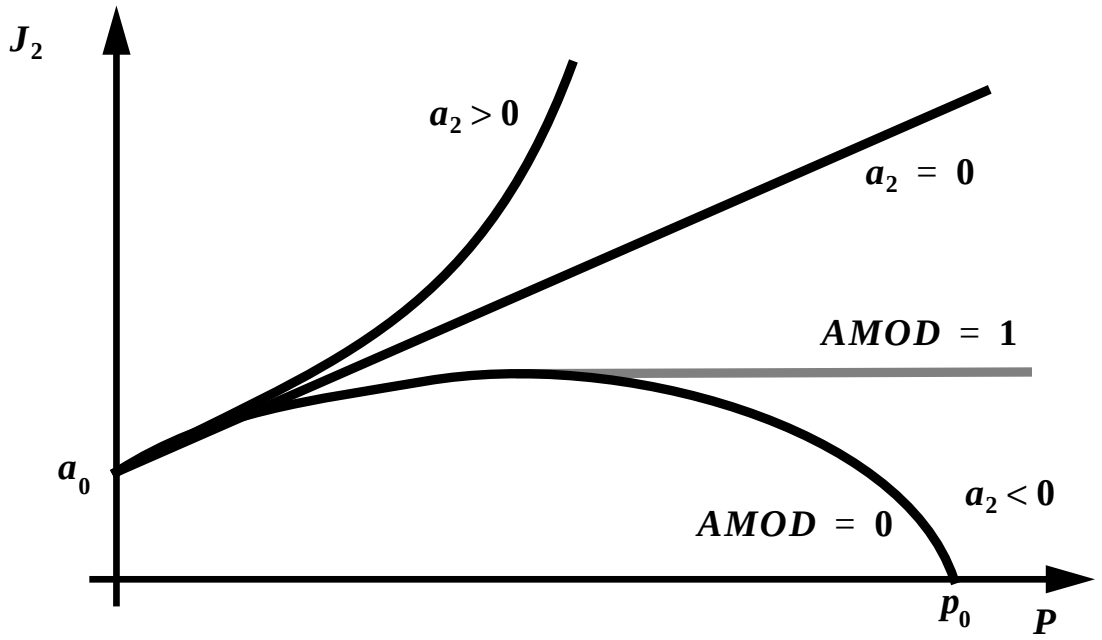


Figure 4.4
Material type 5 failure surface forms in J_2 vs. p space.

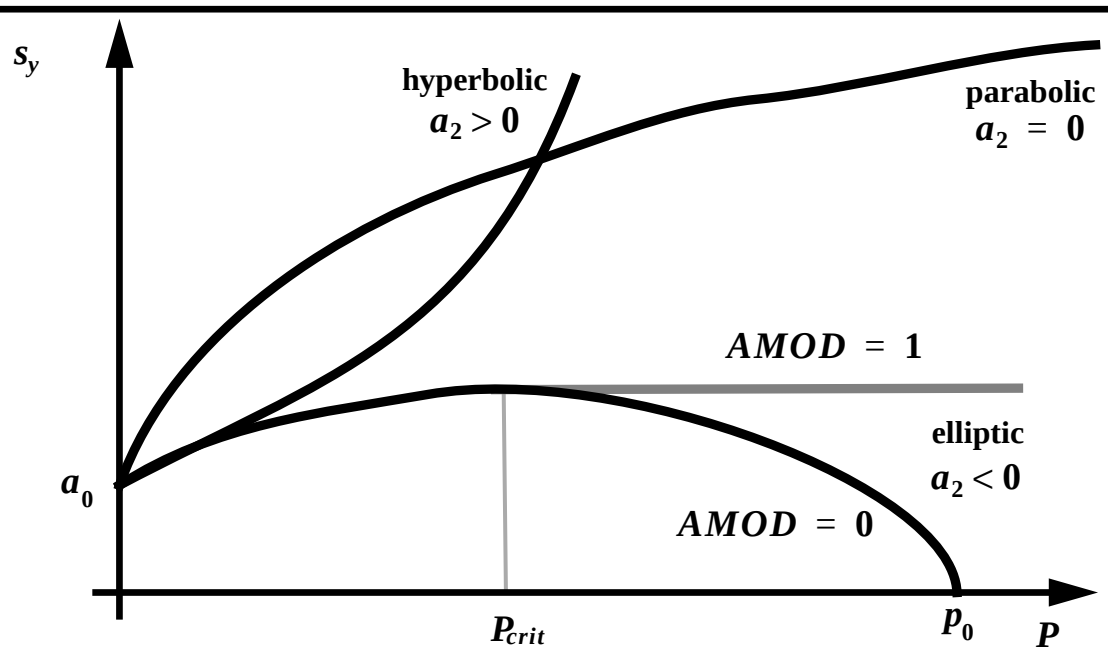


Figure 4.5
Material type 5 failure surface forms in σ_y vs. p space.

The unmodified elliptic yield surface crosses the $\sigma_y = 0$ at a pressure p_0, as shown in Figure 4.5. Thus, p_0 is the maximum pressure at which the unmodified elliptic failure surface may be used:

$$p_0 = \frac{-a_1 + \sqrt{a_1^2 - 4a_0a_2}}{2a_2}. \tag{4-17}$$

The modified elliptic yield surface transitions to a von Mises surface at pressures greater than p_{crit}, where

$$p_{crit} = -\frac{a_1}{2a_2}. \tag{4-18}$$

Material ‘‘erosion’’ and failure may be obtained by defining a nonzero pressure cutoff p_{cut} and specifying that this material is active for automatic contact. If erosion is active, then the material erodes whenever the pressure becomes more tensile than p_{cut}.

5.7 DYNA2D Material Type 6: Viscoelastic

Command	Variable	Description
k	K	Bulk modulus
g0	G_0	Short term shear modulus
gi	G_∞	Long term shear modulus
tp	β	Decay constant

The deviatoric stresses are found from

$$s_{ij} = 2\int_0^t G(t-\tau)\dot{e}_{ij}d\tau\ , \tag{4-19}$$

where the shear relaxation behavior is described by

$$G(t) = G_\infty + (G_0 - G_\infty)e^{-\beta t} \tag{4-20}$$

and $\dot{e}_{ij}$ is the deviatoric strain rate. The volumetric response is elastic, so the pressure p is computed from the current volumetric strain ε_v using

$$p = -K\varepsilon_v, \tag{4-21}$$

where K is the elastic bulk modulus.

5.8 DYNA2D Material Type 7:Blatz-Ko Hyperelastic Rubber

Command	Variable	Description
g	G	Shear modulus

This hyperelastic model is appropriate for materials undergoing moderately large strains. In this formulation, the second Piola-Kirchhoff stress τ is computed as

$$\tau_{ij} = G\left(\frac{1}{\nu}C_{ij} - V^{\frac{-1}{1-2\nu}}\delta_{ij}\right), \tag{4-22}$$

where V is the relative volume, C_{ij} is the right Cauchy-Green strain tensor, and ν is Poisson's ratio which is set to 0.463 internally. The Cauchy stress σ is then found from τ using

$$\sigma = \frac{1}{J}\boldsymbol{F}\tau\boldsymbol{F}^{T}, \tag{4-23}$$

where $\boldsymbol{F}$ is the deformation gradient and J is the Jacobian of the deformation.

5.9 DYNA2D Material Type 8: High Explosive Burn

An equation of state must be used with this model.

Command	Variable	Description
d	D	Detonation velocity
pcj	P_{CJ}	Chapman-Jougeut pressure

This model is used in conjunction with the HE burn option to model the burning of explosives. The detonation velocity D is the velocity of a detonation or burn front. The Chapman-Jouguet pressure P_{CJ} is the maximum pressure realizable in a constant volume adiabatic burn.

During DYNA2D initialization, the lighting time of each element is computed using the selected algorithm. These lighting times may be directly specified in the input, or may be calculated from specified detonation points or lines using the programmed burn options. If detonation points are defined, then the lighting time t_L for an element is computed based on the distance from the center of the element to the nearest detonation point divided by the detonation velocity D.

Burn fractions are computed to control the release of chemical energy for simulating high explosive detonations. If the ''beta burn'' option $IHE = 0$ is selected, then the burn fraction F is computed from

$$F = \beta(1 - V) = \frac{1 - V}{1 - V_{CJ}}, \tag{4-24}$$

where V is the current relative volume, V_{CJ} is the Chapman-Jouget relative volume, and

$$\beta = \frac{1}{1 - V_{CJ}}. \tag{4-25}$$

For other burn options, the burn fraction is computed from

$$F = max(F_1, F_2), \tag{4-26}$$

where

$$F_1 = ((t - t_L)D)/(1.5h) \tag{4-27}$$

if $t > t_L$, and $F_1 = 0$ if $t < t_L$, and h is a characteristic dimension of the element under consideration. F_2 is computed from

$$F_2 = \beta(1 - V) = \frac{1 - V}{1 - V_{CJ}}. \tag{4-28}$$

If the above equations produce a burn fraction that is greater than one, then it is reset to one.

The burn front propagates by multiplying the pressure computed from an equation-of-state by the current burn fraction,

$$p = F\,p_{EOS}(V, E), \tag{4-29}$$

where $p_{EOS}(V, E)$ is the pressure computed from the equation-of-state at the current relative volume V and energy E. High explosives typically have large initial internal energies, E_0, which yield large pressures as $F \rightarrow 1$.

5.10 DYNA2D Material Type 9: Fluid

An equation of state must be used with this model.

Command	Variable	Description
pc	p_{cut}	Pressure cutoff
mu	μ	Viscosity coefficient

The fluid material has no stiffness, and must be used with an equation-of-state. A viscous stress is computed from

$$s_{ij} = \mu \dot{e}_{ij}, \tag{4-30}$$

where $\dot{e}_{ij}$ is the deviatoric strain rate and s_{ij} is the deviatoric stress. Materials with no viscosity may reach large distortions under very small shear loads, so a nonzero viscosity should always be used.

The pressure cutoff, p_{cut}, is negative in tension. If the pressure becomes more tensile than p_{cut}, then it is reset to that value. Thus, the pressure cutoff can be interpreted as an approximate model of cavitation. The deviatoric stresses arising from viscous effects are unaffected by the tensile pressure cutoff.

5.11 DYNA2D Material Type 10: Isotropic-Elastic-Plastic-Hydrodynamic

An equation of state must be used with this model.

Command	Variable	Description
g	G	Shear modulus
sigy	σ_o	Yield stress
ep	E_p	Plastic modulus
pc	P_{cut}	Pressure cutoff
a1	a_1	Linear pressure hardening coefficient
a2	a_2	Quadratic pressure hardening coefficient
spall	*spall*	Spall model
epsf	$\bar{\varepsilon}^p{}_f$	Effective plastic strain at failure
npts	n	Number of points in stress-effective plastic strain curve
eps	$\varepsilon_1 \ldots \varepsilon_n$	Effective plastic strain
es	$\sigma_1 \ldots \sigma_n$	Effective yield stress
e0	*E0*	Equivalent plastic strain at failure

If a tabulated yield stress vs. effective plastic strain curve is not given, then the initial yield stress σ_0, plastic hardening modulus E_p, and pressure hardening coefficients a_1 and a_2 are used. In this case, a pressure hardening bilinear stress-strain curve similar to that shown in Figure 4.2 is obtained with linear isotropic strain hardening ($\beta = 1.0$).

The yield condition can be written

$$\phi = \bar{\sigma} - \sigma_y(\bar{\varepsilon}^p, p), \tag{4-31}$$

where $\bar{\sigma}$ is the effective stress and σ_y is the current yield stress, which may be a function of the effective plastic strain $\bar{\varepsilon}^p$ and pressure p. The effective stress $\bar{\sigma}$ is given by

$$\bar{\sigma} = \left(\frac{3}{2} s_{ij} s_{ij}\right)^{\frac{1}{2}}, \tag{4-32}$$

where s_{ij} is the deviatoric stress tensor.

The hardening law has the form

$$\sigma_y = \sigma_0 + E_p\bar{\varepsilon}^p + (a_1 + a_2 p)\hat{p}, \tag{4-33}$$

where p is the pressure (positive in compression), and $\hat{p}$ is the tension-limited pressure found from

$$\hat{p} = max(p, 0)\,. \tag{4-34}$$

The effective plastic strain $\bar{\varepsilon}^p$ is given by

$$\bar{\varepsilon}^p = \int_0^t d\bar{\varepsilon}^p, \tag{4-35}$$

where $d\bar{\varepsilon}^p$ is the incremental effective plastic strain. The plastic modulus can be related to Young's modulus E and the tangent modulus E_T using

$$E_p = \frac{EE_T}{E - E_T}. \tag{4-36}$$

The plastic modulus E_P is the slope of the inelastic portion of the effective stress $\bar{\sigma}$ vs. effective plastic strain $\bar{\varepsilon}^p$ curve, and the tangent modulus E_T is the slope of the inelastic part of a uniaxial stress vs. strain curve (or equivalently, the effective stress vs. effective strain curve).

If tabulated values of yield stress vs. effective plastic strain are specified, a nonlinear strain hardening curve like that shown in Figure 4.6 may be defined. In this case the plastic hardening modulus and pressure hardening coefficients are not used, and the yield stress is given as

$$\sigma_y = f(\bar{\varepsilon}^p), \tag{4-37}$$

where $f(\bar{\varepsilon}^p)$ is interpolated from the specified yield stress vs. effective plastic strain curve. Any number of points, from 2 to 16, may be used to define the hardening curve. This option permits additional detail to be included in the nonlinear strain hardening law, but pressure hardening is not modeled with this approach.

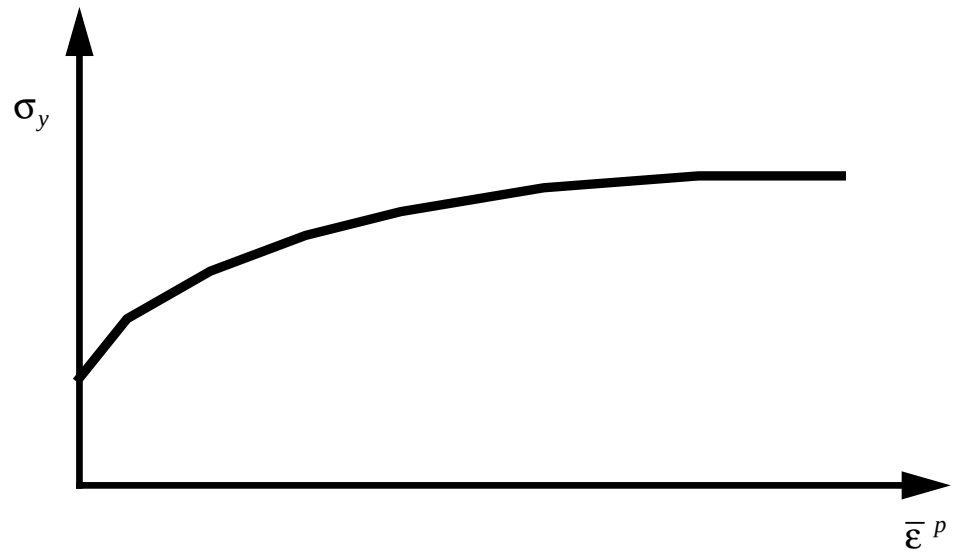


Figure 4.6
Tabulated yield stress vs. effective plastic strain curve.

Three spall models are offered to represent material splitting, cracking, and failure under tensile loads. The pressure limit model, *SPALL = 1*, limits the hydrostatic tension to the specified value, p_{cut}. If pressures more tensile than p_{cut} are calculated, the pressure is reset to p_{cut}. This option is not strictly a spall model, since the deviatoric stresses are unaffected by the pressure reaching the tensile cutoff, and the pressure cutoff value p_{cut} remains unchanged throughout the analysis. The maximum principal stress model, *SPALL = 2*, detects spall if the maximum (most tensile) principal stress σ_{max} exceeds the limiting value $-p_{cut}$. Note: the negative sign is required because p_{cut} is measured positive in compression, while σ_{max} is positive in tension. Once spall is detected, the deviatoric stresses are set to zero, and no hydrostatic tension $(p < 0)$ is permitted. If tensile pressures are calculated, they are reset to zero in the spalled material. Thus, the spalled material behaves as a rubble or incohesive material. The hydrostatic tension model, *SPALL = 3*, detects spall if the pressure becomes more tensile than the specified limit, p_{cut}. Once spall is detected the deviatoric stresses are set to zero, and the pressure is required to be compressive. If hydrostatic tension $(p < 0)$ is subsequently calculated, the pressure is reset to zero for that element.

Material ‘‘erosion’’ and failure may be obtained by defining a nonzero effective plastic strain at failure $\bar{\varepsilon}_f^p$ and specifying that this material is active for automatic contact.

5.12 DYNA2D Material Type 11: Steinberg-Guinan High Rate Elastic-Plastic

An equation of state must be used with this model.

Command	Variable	Description
g0	G_0	Shear modulus constant
sig0	σ_0	Yield stress constant
beta	β	Strain hardening law constant
n	n	Strain hardening exponent
gama	γ_i	Initial plastic strain
sigm	σ_m	Yield stress work hardening limit
b	b	Shear modulus pressure constant
bp	b'	Yield stress pressure constant
h	h	Energy coefficient
f	f	Energy exponential coefficient
a	a	Atomic weight; if a = 0, R' must be defined
t0	T_{mo}	Melting temperature constant
gam0	γ_0	Thermodynamic gamma
sa	α	Thermodynamic constant
pc	P_{cut}	Pressure cutoff
trm	T_{room}	Room temperature
deby	Θ	Debye coefficient
spall	*spall*	Spall model flag: EQ. 0.0: default set to 2.0 EQ. 1.0: Pressure limit model EQ. 2.0: Maximum principal stress spall criterion EQ. 3.0: Hydrostatic tension spall criterion
rp	R'	NOTE: If $R' \neq 0$, atomic weight is not used
epsf	ε^{-p}_f	Effective plastic strain at failure
nfit	*nfit*	Polynomial order for fit; $1 \leq$ nfit ≤ 10
ivar	*flag*	Cold compression energy polynomial flag: EQ. 0.0: Polynomial coefficients given or fit in terms of η EQ. 1.0: Polynomial coefficients given or fit in terms of μ

min	*min_limit*	Optional minimum limit for energy fit. Input η_{min} if ivar = 0 Input μ_{min} if ivar = 0
max	*max_limit*	Optional maximum limit for energy fit. Input η_{max} if ivar = 0 Input μ_{max} if ivar = 0
ec0 ... ec9	$EC_0 \ldots EC_9$	Cold compression polynomial coefficients

In terms of the foregoing input parameters, we define the shear modulus, G, before the material melts as:

$$G = G_0\left[1 + bpV^{\frac{1}{3}} - h\left(\frac{E_i - E_c}{3R'} - 300\right)\right]e^{-\frac{fE_i}{E_m - E_i}}, \tag{4-38}$$

where p is the pressure, V is the relative volume, E_i is the current energy, E_c is the cold compression energy, and E_m is the melting energy. The cold compression energy is calculated using

$$E_c(x) = \int_0^x p\,dx, \tag{4-39}$$

where $x = 1 - V$. The equation is integrated using initial energy E_o and pressure P_o conditions that correspond to zero K and are given by

$$E_o = -3R'T_{room} \tag{4-40}$$

and

$$P_o = \gamma_o E_o \mathrm{Debye}\left(\frac{\theta}{T_{room}}\right). \tag{4-41}$$

Here Debye is the Debye correction factor, and has a default value of 1 when $\theta = 0$. The melting energy is found from the cold compression energy and the melting temperature using

$$E_m(x) = E_c(x) + 3R'T_m(x), \tag{4-42}$$

where the melting temperature T_m is given by

$$T_m(x) = \frac{T_{mo} exp(2ax)}{V^{2\left(\gamma_0 - a - \frac{1}{3}\right)}} \tag{4-43}$$

and T_{mo} is the melting temperature at the initial density, ρ_0.

In the above equations, R' is defined by

$$R' = \frac{R\rho}{A}, \tag{4-44}$$

where R is the universal gas constant and A is the atomic weight. **Note that if R' is not defined, DYNA2D computes it with R in the cm-gram-microsecond system of units.** Thus, this option should not be used unless the entire analysis model is defined in the cm-gram-microsecond second system of units.

If E_m exceeds E_i (i.e., the material has not melted), then the yield strength σ_y is given by:

$$\sigma_y = \sigma_0' \left[1 + b' p V^{\frac{1}{3}} - h\left(\frac{E_i - E_c}{3R'} - 300\right)\right] e^{-\frac{fE_i}{E_m - E_i}} \quad . \tag{4-45}$$

The work-hardened yield stress σ'_0 is found from the initial yield stress σ_0 and the accumulated effective plastic strain $\bar{\varepsilon}^p$ using the hardening law

$$\sigma'_0 = \sigma_0[1 + \beta(\gamma_i + \bar{\varepsilon}^p)]^n , \tag{4-46}$$

where γ_i is the initial plastic strain. If the work-hardened yield stress σ_0 exceeds the limiting value σ_m, then σ'_0 is reset to σ_m. After the materials melts ($E_i > E_m$), the yield stress σ_y and shear modulus G are reset to one half their initial value.

The evaluation of the cold compression energy $E_c(x)$ using (4-39) is approximated during execution. The independent variable is chosen as η, and the polynomial takes the form

$$E_c = \sum_{i=0}^{9} EC_i \eta^i . \tag{4-47}$$

Note that the density and compression variables are related by

$$x = 1 - V = \frac{\mu}{\mu + 1} = 1 - \frac{1}{\eta} . \tag{4-48}$$

If the coefficients EC_0 through EC_9 are not specified in the input, DYNA2D will fit the cold compression energy with up to a ten term polynomial expansion using a least squares method. If the order of the polynomial is not specified, DYNA2D will automatically pick the best polynomial order that fits the EOS generated data. Otherwise, DYNA2D will attempt to fit the data to the polynomial order desired.

A Debye correction can be applied to the cold compression energy to improve the model's temperature response. This option is activated by specifying a non-zero value of the Debye coefficient θ.

A choice of three spall models is offered to represent material splitting, cracking, and failure under tensile loads. The pressure limit model, *SPALL = 1*, limits the hydrostatic tension to the specified value, p_{cut}. If pressures more tensile than this limit are calculated, the pressure is reset to p_{cut}. This option is not strictly a spall model, since the deviatoric stresses are unaffected by the pressure reaching the tensile cutoff, and the pressure cutoff value p_{cut} remains unchanged throughout the analysis. The maximum principal stress spall model, *SPALL = 2*, detects spall if the maximum (most tensile) principal stress σ_{max} exceeds the limiting value $-p_{cut}$. Note that the negative sign is required because p_{cut} is measured positive in compression, while σ_{max} is positive in tension. Once spall is detected with this model, the deviatoric stresses are set to zero, and no hydrostatic tension $(p < 0)$ is permitted. If tensile pressures are calculated, they are reset to zero in the spalled material. Thus, the spalled material behaves as a rubble or incohesive material. The hydrostatic tension spall model, *SPALL = 3*, detects spall if the pressure becomes more tensile than the specified limit, p_{cut}. Once spall is detected the deviatoric stresses are set to zero, and the pressure is required to be compressive. If hydrostatic tension $(p < 0)$ is subsequently calculated, the pressure is reset to zero for that element.

Material ''erosion'' and failure may be obtained by defining a nonzero effective plastic strain at failure $\bar{\varepsilon}_f^p$ and specifying that this material is active for automatic contact.

5.13 DYNA2D Material Type 12: Johnson / Cook Elastic-Plastic

An equation of state must be used with this model.

Command	Variable	Description
g	G	Shear modulus
a	a	Yield stress constant
b	B	Strain hardening constant
beta	β	Strain hardening law constant
h	n	Strain hardening exponent
c	C	Strain rate dependence coefficient
m	m	Temperature dependence exponent
tm	T_m	Melt temperature (˚K)
tr	T_r	Room temperature (˚K)
e0	ε_0	Reference strain rate
sh	c_v	Specific temperature
pcut or sigm	p_{cut} or σ_m	Pressure cutoff or failure stress
spall	*spall*	Spall model flag: EQ. 1.0: Pressure limit model EQ. 2.0: Maximum principal stress spall criterion EQ. 3.0: Hydrostatic tension spall criterion
psif	*flag*	Plastic strain iteration flag: EQ. 0.0: Fast approximation solution for plastic strain (default) EQ. 1.0: Accurate iterative solution for plastic strain (more expensive than default)
epsf	ε^{-p}_f	Effective plastic strain at failure
d1	*D1*	First failure parameter
d2	*D2*	Second failure parameter
d3	*D3*	Third failure parameter
d4	*D4*	Fourth failure parameter
d5	*D5*	Fifth failure parameter

The yield stress is written as

$$\sigma_y = [A + B(\bar{\varepsilon}^p)^n][1 + C ln(\dot{\varepsilon}^*)][1 - (T^*)^m] \quad , \tag{4-49}$$

where A, B, C, n and m are input constants, $\bar{\varepsilon}^p$ is the effective plastic strain, $\dot{\varepsilon}^*$ is the nondimensional strain rate, and T^* is the homologous temperature. The effective plastic strain $\bar{\varepsilon}^p$ is given by

$$\bar{\varepsilon}^p = \int_0^t d\bar{\varepsilon}^p, \tag{4-50}$$

where $d\bar{\varepsilon}^p$ is the incremental effective plastic strain.
The nondimensional strain rate $\dot{\varepsilon}^*$ is calculated from

$$\dot{\varepsilon}^* = \frac{\dot{\bar{\varepsilon}}^p}{\dot{\varepsilon}_0} \tag{4-51}$$

where $\dot{\bar{\varepsilon}}^p$ is the effective plastic strain rate and $\dot{\varepsilon}_0$ is the reference strain rate defined in the input. The homologous temperature T^* is the ratio of the current temperature to the melting temperature when both are expressed in degrees Kelvin. Temperature change in this model is computed assuming adiabatic conditions, i.e., no heat transfer between elements. Heat is generated in an element by plastic work, and the resulting temperature rise is computed using the specific heat for the material.

This implementation of the Johnson-Cook model also contains a damage model. The strain at fracture ε_f is given by

$$\varepsilon_f = [D_1 + D_2 exp(D_3\sigma^*)][1 + D_4 ln(\dot{\varepsilon}^*)][1 + D_5 T^*] \tag{4-52}$$

where σ^* is the ratio of pressure divided by effective stress

$$\sigma^* = \frac{p}{\bar{\sigma}}, \tag{4-53}$$

and effective stress $\bar{\sigma}$ is found from

$$\bar{\sigma} = \left(\frac{3}{2} s_{ij} s_{ij}\right)^{\frac{1}{2}} . \tag{4-54}$$

In this equation, "*s*" represents the deviatoric stress. Note that this definition of σ^* **is** reversed in sign from convention in the original publications of Johnson and Cook; the sign of D_3 should be chosen carefully.

Fracture occurs when the damage parameter D exceeds the value of 1. The evolution of the damage parameter is given by

$$D = \sum \frac{\Delta \bar{\varepsilon}^p}{\varepsilon_f}, \tag{4-55}$$

where the summation is performed over all time steps in the analysis. When fracture occurs, all stresses are set to zero and remain zero for the rest of the calculation.

A choice of three spall models is offered to represent material splitting, cracking, and failure under tensile loads. The pressure limit model, *SPALL = 1*, limits the hydrostatic tension to the specified value, p_{cut}. If pressures more tensile than this limit are calculated, the pressure is reset to p_{cut}. This option is not strictly a spall model, since the deviatoric stresses are unaffected by the pressure reaching the tensile cutoff, and the pressure cutoff value p_{cut} remains unchanged throughout the analysis. The maximum principal stress spall model, *SPALL = 2*, detects spall if the maximum (most tensile) principal stress σ_{max} exceeds the limiting value σ_m. Once spall is detected with this model, the deviatoric stresses are set to zero, and no hydrostatic tension $(p < 0)$ is permitted. If tensile pressures are calculated, they are reset to 0 in the spalled material. The hydrostatic tension spall model, *SPALL = 3*, detects spall if the pressure becomes more tensile than the specified limit, p_{cut}. Once spall is detected, the deviatoric stresses are set to zero and the pressure is required to be compressive. If hydrostatic tension $(p < 0)$ is calculated, then the pressure is reset to 0 for that element.

Material ‘‘erosion’’ and failure may be obtained by defining a nonzero effective plastic strain at failure $\bar{\varepsilon}_f^p$ and specifying that this material is active for automatic contact. Alternatively, by specifying a positive FSD and nonzero damage parameters, material ‘‘erosion’’ will be controlled by the accumulated damage parameter D. Both material erosion methods can be used simultaneously. In this case which every condition is satisfied first will determine erosion.

5.14 DYNA2D Material Type 13: Power Law Isotropic Elastic-Plastic

Command	Variable	Description
e	E	Young's modulus
v	ν	Poisson's ratio
k	K	Strength coefficient
n	N	Hardening exponent
epsf	ε^{-p}_f	Effective plastic strain at failure

The material behavior is elastoplastic with nonlinear isotropic strain hardening given by a power law expression. The yield condition can be written

$$\phi = \bar{\sigma} - \sigma_y(\bar{\varepsilon}^p) , \tag{4-56}$$

where $\bar{\sigma}$ is the effective stress and σ_y is the current yield stress. The hardening law has the form

$$\sigma_y = k(\varepsilon_0 + \bar{\varepsilon}^p)^n , \tag{4-57}$$

where ε_0 is the initial yield strain given by

$$\varepsilon_0 = \left(\frac{E}{k}\right)^{\frac{1}{n-1}} . \tag{4-58}$$

Material ‘‘erosion’’ and failure may be obtained by defining a nonzero effective plastic strain at failure $\bar{\varepsilon}^p_f$ and specifying that this material is active for automatic contact.

5.15 DYNA2D Material Type 14: Viscoplastic

Command	Variable	Description
e	E	Young's modulus
v	ν	Poisson's ratio
t	T_0	Initial temperature (˚K)
rcv	ρC_v	Density specific heat
b	β	Hardening parameter
c1	C_1	Rate dependent yield stress coefficient
c2	C_2	Rate dependent yield stress exponent
c3	C_3	Rate independent yield stress coefficient
c4	C_4	Rate independent yield stress exponent
c5	C_5	Transition coefficient
c6	C_6	Transition exponent
c7	C_7	Hardening coefficient
c8	C_8	Hardening exponent
c9	C_9	Dynamic recovery coefficient
c10	C_{10}	Dynamic recovery exponent
c11	C_{11}	Diffusion recovery coefficient
c12	C_{12}	Diffusion recovery exponent

This model represents a modified implementation of a unified creep plasticity model proposed by Bamman (1984).

The history dependence of this model is characterized through the introduction of two internal state variables, a scalar κ, and a second order tensor α. The governing constitutive equations are of the form

$$\dot{\sigma} = \frac{E\nu}{(1+\nu)(1-2\nu)} tr(\dot{\varepsilon}) 1 + \frac{E}{1+\nu}(\dot{\varepsilon} - \dot{\varepsilon}^p) \quad , \tag{4-59}$$

$$\dot{\varepsilon}^p = f(T) \sinh\left[\frac{|\eta| - \kappa - Y(T)}{V(T)}\right] \frac{\eta}{|\eta|} \, , \tag{4-60}$$

$$\dot{\alpha} = k(T)(1-\beta)\dot{\varepsilon}^p - \frac{(g(T)+h(T)|\dot{\boldsymbol{e}}^p|)|\alpha|\alpha}{1-\beta} \quad , \tag{4-61}$$

$$\dot{\kappa} = k(T)\beta|\dot{\boldsymbol{e}}^p| - \frac{(g(T)+h(T)|\dot{\boldsymbol{e}}^p|)\kappa^2}{\beta} \quad , \tag{4-62}$$

where the translated stress η is given by

$$\eta = \mathbf{s} - \alpha, \tag{4-63}$$

and **s** is the deviatoric stress. The inelastic behavior of the model is governed by six temperature dependent parameter functions *V(T), Y(T), f(T), h(T), k(T),* and *g(T)*. Each of the functions is an exponential and defined by two material parameters as follows:

$$V(T) = C_1 e^{-C_2/T} \quad \text{(rate dependent yield stress)}, \tag{4-64}$$

$$Y(T) = C_3 e^{C_4/T} \quad \text{(rate independent yield stress)}, \tag{4-65}$$

$$f(T) = C_5 e^{-C_6/T} \quad \text{(transition to rate dependent behavior)}, \tag{4-66}$$

$$h(T) = C_7 e^{-C_8/T} \quad \text{(hardening)}, \tag{4-67}$$

$$k(T) = C_9 e^{-C_{10}/T} \quad \text{(dynamic recovery)}, \tag{4-68}$$

$$g(T) = C_{11} e^{-C_{12}/T} \quad \text{(diffusion controlled static or thermal recovery)}. \tag{4-69}$$

The specification of hardening parameter β, where $0.0 \le \beta \le 1.0$, results in either kinematic, isotropic, or a combination of kinematic and isotropic hardening. Purely kinematic or purely isotropic hardening is obtained by setting $\beta = 0.0$ or $\beta = 1.0$, respectively. For these cases, numerical perturbations are used to prevent the governing equations from becoming singular. The model also accounts for adiabatic heating due to plastic work. Temperature rate $\dot{T}$ is defined in terms of the density ρ and specific heat C_v as

$$\dot{T} = \frac{0.95}{\rho C_v}\sigma\dot{\varepsilon}^p \quad . \tag{4-70}$$

5.16 DYNA2D Material Type 15:Generalized Armstrong-Zerilli Elastic Plastic

An equation of state must be used with this model.

Command	Variable	Description
g0	G_0	Shear modulus at reference temperature
ixtal	*flag*	Crystal geometry flag: EQ. -1: BCC EQ. +1: FCC
l	L	Polycrystal grain diameter
sigg	$\Delta\sigma_G$	Yield stress constant
B0	B_0	Temperature coefficient
bet0	β_0	Temperature exponent
bet1	β_1	Strain rate thermal exponent
k0	K_0	Strain hardening coefficient
n	n	Strain hardening exponent
k1	K_1	Strength constant
ke	K_ε	Grain size coefficient
a0	a_0	Shear modulus ratio constant
a1	a_1	Shear modulus ratio coefficient
b0	b_0	Specific heat constant
b1	b_1	Specific heat coefficient
pcut	p_{cut}	Pressure cutoff (negative in tension)
epsf	ε^{-p}_f	Effective plastic strain at failure
T0	T_0	Initial temperature

The Armstrong-Zerilli model is a strain rate and temperature dependent elastic plastic model for metals undergoing large strains over a wide range of strain rates. The yield function has two forms depending on the crystal structure of the material. For BCC materials ($IXTAL = -1$), the yield function may be written as

$$\sigma_y = \Delta\sigma_G + B_0 exp[(-\beta_0 + \beta_1 ln(\dot{\bar{\varepsilon}}))T] + [K_0(\bar{\varepsilon}^p)^n + K_1]\frac{\mu_T}{\mu_0} + K_\varepsilon L^{-1/2} \quad , \tag{4-71}$$

and for FCC materials ($IXTAL = 1$) the yield function becomes

$$\sigma_y = \Delta\sigma_G + B_0\sqrt{\bar{\varepsilon}^p}exp[(-\beta_0 + \beta_1 ln(\dot{\bar{\varepsilon}}))T] + [K_0(\bar{\varepsilon}^p)^n + K_1]\frac{\mu_T}{\mu_0} + K_\varepsilon L^{-1/2} \quad , \tag{4-72}$$

where $\bar{\varepsilon}^p$ is effective plastic strain, $\dot{\bar{\varepsilon}}$ is the effective strain rate, μ_T is the shear modulus at the current temperature T, μ_0 is the shear modulus at the initial temperature T_0, and $\Delta\sigma_G$, B_0, β_0, β_1, K_0, K_1, and K_ε are material constants, and L is the grain size of the material. The shear modulus ratio is approximated as a linear function of temperature,

$$\frac{\mu_T}{\mu_0} = a_0 + a_1 T \tag{4-73}$$

where a_0 and a_1 are constants. The specific heat is approximated as a linear function of temperature,

$$c_p(T) = b_0 + b_1 T \, , \tag{4-74}$$

where b_0 and b_1 are constants.

The pressure cutoff feature limits the hydrostatic tension to the specified value, p_{cut}. If pressures more tensile than this limit are calculated, the pressure is reset to p_{cut}.

Material ‘‘erosion’’ and failure may be obtained by defining a nonzero effective plastic strain at failure $\bar{\varepsilon}_f^p$ and specifying that this material is active for automatic contact.

5.17 DYNA2D Material Type 16: Concrete / Geologic Material

An equation of state type 8, 9, or 11 must be used with this model.

Command	Variable	Description
ec1	v or $-G$	Poisson's ratio, v, for constant v model or negative of shear modulus, $-G$, for constant G model
sigc	σ_{cut}	Maximum principal stress at failure
a0	a_0	Cohesion
a1	a_1	Pressure hardening coefficient
a2	a_2	Pressure hardening coefficient
b1	b_1	Damage scaling factor
a0f	a_{0f}	Cohesion for failed material
a1f	a_{1f}	Pressure hardening coefficient for failed material
fr	f_r	Percent reinforcement: $0 \le f_r \le 100\%$
er	E_r	Elastic modulus for reinforcement
vr	v_r	Poisson's ratio for reinforcement
sig0	σ_0	Initial yield stress
et	E_T	Tangent modulus
n1	N_1	Load curve giving rate sensitivity for principal material
n2	N_2	Load curve giving rate sensitivity for rate reinforcement
npts	*npts*	Number of effective plastic strain (or pressure) and yield stress tabulations
epsp	$\varepsilon^{-p}{}_1 \ldots \varepsilon^{-p}{}_n$ or $p_1 \cdots p_n$	Effective plastic strains or pressures
sigy	$\sigma_{y1} \ldots \sigma_{yn}$	Yield stresses

Material Type 16 can be used in two major modes - a simple tabular pressure-dependent yield surface and a complex model featuring two yield versus pressure functions with various means of migrating from one curve to the other. For both modes, load curve N_1 is a strain rate multiplier for the yield strength.

Response Mode I. Tabulated Yield Stress Versus Pressure

This mode is well suited for implementing standard geologic models, e.g., Mohr-Coulomb yield surface with a Tresca limit (Figure 4.7). Under conventional triaxial compression conditions, DYNA2D requires an ordinate of $\sigma_1 - \sigma_3$ rather than the more widely used $\frac{\sigma_1 - \sigma_3}{2}$, where σ_1 is the maximum principal stress and σ_3 is the minimum principal stress.

To invoke Mode I of this model, set a_0, a_1, a_2, b_1, a_{0f}, and a_{1f} to zero. The tabulated values of pressure and the corresponding values of yield stress should be specified. The parameters relating to reinforcement properties, initial yield stress, and tangent modulus are not used in this response mode, and should be set to zero.

Simple tensile failure

Note that a_{1f} is reset internally to 1/3 even though it is input as zero; this defines a failed material curve of slope $3p$, where p denotes pressure (positive in compression). In this case the yield strength is taken from the tabulated yield vs. pressure curve until the maximum principal stress (σ_1) in the element exceeds the tensile cut-off (σ_{cut}). For every time step that $\sigma_1 > \sigma_{cut}$ the yield strength is scaled back by a fraction of the distance between the two curves until after 20 time steps the yield strength is defined by the failed curve. The only way to inhibit this feature is to set σ_{cut} arbitrarily large.

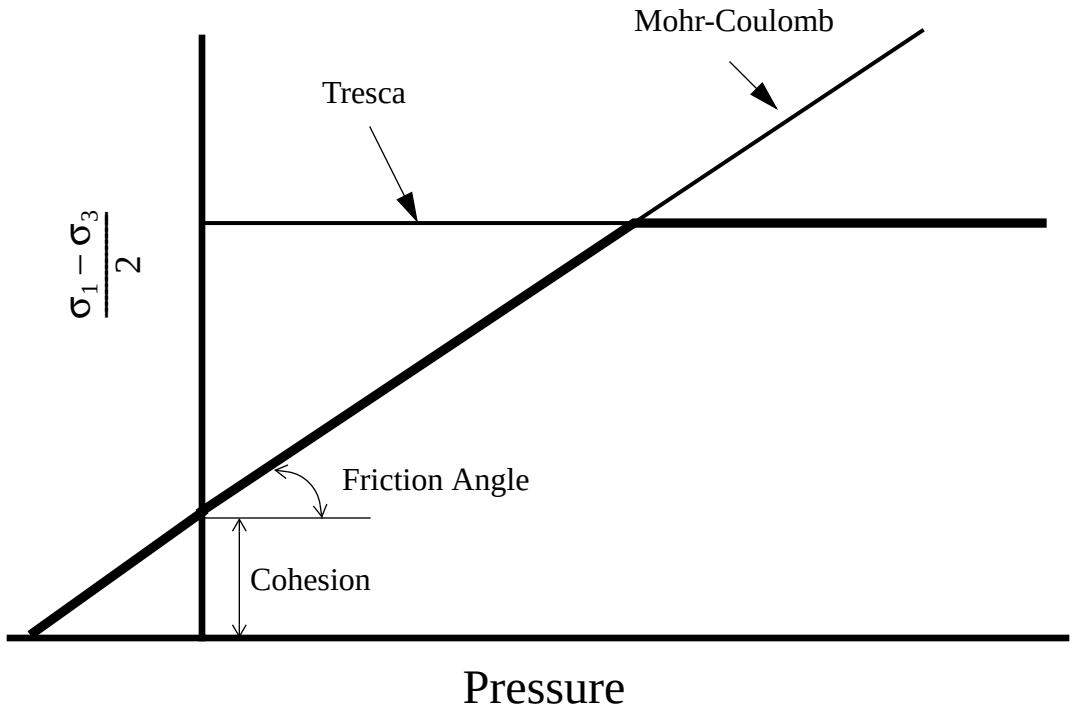


Figure 4.7
Mohr-Coulomb surface with a Tresca limit.

Response Mode II. Two Curve Model with Damage and Failure

This approach uses two yield versus pressure curves of the form

$$\sigma_y = a_0 + \frac{p}{a_1 + a_2 p}. \tag{4-75}$$

The upper curve is best described as the maximum yield strength curve and the lower curve is the failed material curve. There are a variety of ways of moving between the two curves, and each is discussed below.

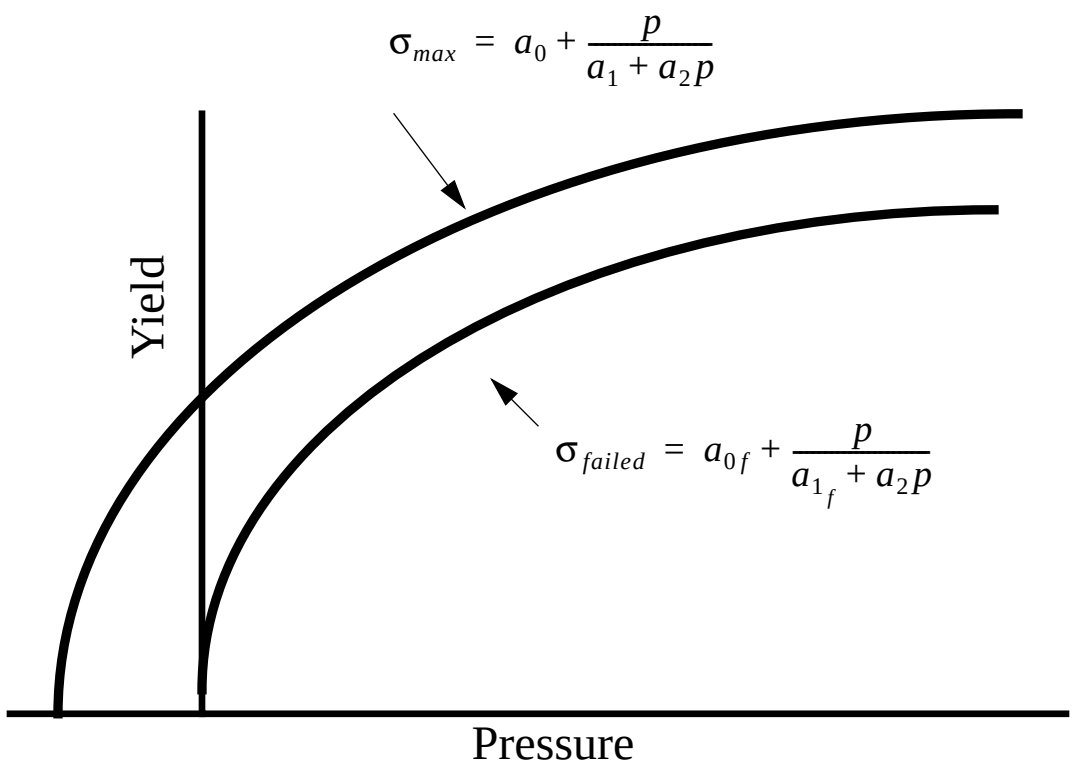


Figure 4.8
Two-curve concrete model with damage and failure.

Mode II.A: Simple tensile failure

Define a_0, a_1, a_2, a_{0f} and a_{1f}, and set b_1 to zero. The yield strength is taken from the maximum yield curve until the maximum principal stress (σ_1) in the element exceeds the tensile cut-off (σ_{cut}). For every time step that $\sigma_1 > \sigma_{cut}$ the yield strength is scaled back by a fraction of the distance between the two curves until after 20 time steps the yield strength is defined by the failed curve.

Mode II.B: Tensile failure plus plastic strain scaling

Define a_0, a_1, a_2, a_{0f} and a_{1f}, set b_1 to zero, and define a scale factor, η, versus effective plastic strain. DYNA2D evaluates η at the current effective plastic strain and then calculates the yield stress as

$$\sigma_{yield} = \sigma_{max} - \eta(\sigma_{max} - \sigma_{failed}), \tag{4-76}$$

where σ_{max} and σ_{failed} are found as shown in Figure 4.8.
This yield strength is then subject to scaling for tensile failure as described above. This type of model allows the description of a strain hardening and/or softening material such as concrete.

Mode II.C: Tensile failure plus damage scaling

The change in yield stress as a function of plastic strain arises from physical mechanisms such as internal cracking, and the extent of this cracking is affected by the hydrostatic pressure when the cracking occurs. This mechanism gives rise to the "confinement" effect on concrete behavior. To account for this phenomenon a "damage" function was defined and incorporated into Material Type 16. The damage function is given the form

$$\lambda = \int_0^{\bar{\varepsilon}^p} \frac{d\bar{\varepsilon}^p}{\left(1 + \frac{p}{\sigma_{cut}}\right)^{b_1}} \tag{4-77}$$

Define a_0, a_1, a_2, a_{0f}, a_{1f}, and b_1. η is now given as a function of λ and scales the yield stress as

$$\sigma_{yield} = \sigma_{max} - \eta(\sigma_{max} - \sigma_{failed}) \tag{4-78}$$

and then apply any tensile failure criteria.

Mode II Concrete Model Options

Material Type 16 in Mode II provides for the automatic internal generation of a simple "generic" model for concrete. If a_0 is negative, then σ_{cut} is assumed to be the unconfined concrete compressive strength (f_c') and $-a_0$ is assumed to be a conversion factor from DYNA pressure units to psi. In this case the parameter values generated internally are:

$$\sigma_{cut} = 1.7\left(\frac{(f_c')^2}{-a_0}\right)^{1/3} \tag{4-79}$$

$$a_0 = \frac{f_c'}{4} \tag{4-80}$$

$$a_1 = \frac{1}{3} \tag{4-81}$$

$$a_2 = \frac{1}{3f_c'} \tag{4-82}$$

$$a_{0f} = 0 \tag{4-83}$$

$$a_{1f} = 0.385 \tag{4-84}$$

Note that these a_{0f} and a_{1f} defaults will be overridden by nonzero entries for them. If plastic strain or damage scaling is desired, η and b_1 should be specified in the input. When a_0 is input as a negative quantity, the Equation-of-State can be given as 0 and a trilinear EOS Type 8 model will be automatically generated from the unconfined compressive strength and Poisson's ratio. The EOS 8 model is a simple pressure versus volumetric strain model with no internal energy terms, and should give reasonable results for pressures up to 5 kbar (approximately 75,000 psi).

Mixture model

A reinforcement fraction, $\hat{f}_r$, can be defined along with properties of the reinforcement material. The bulk modulus, shear modulus, and yield strength are then calculated from a simple mixture rule. This feature is *experimental* and should be used with caution. It gives an isotropic effect in the material instead of the true anisotropic material behavior. A reasonable approach would be to use mixture elements only where the reinforcing exists and plain elements elsewhere. When the mixture model is being used, the strain rate multiplier for the principal material is taken from load curve N_1 and the multiplier for the reinforcement is taken from load curve N_2.

5.18 DYNA2D Material Type 18: Extended Two Invariant Geologic Cap Model

Command	Variable	Description
k	K	Initial bulk modulus
g	G	Initial shear modulus
alpha	α	Failure envelope parameter
theta	Θ	Failure envelope linear coefficient
gamma	γ	Failure envelope exponential coefficient
b	β	Failure envelope exponent
r	R	Cap surface axis ratio
d	D	Hardening law exponent
w	W	Hardening law coefficient
x0	X_0	Hardening law parameter
cbar	$\bar{c}$	Kinematic hardening coefficient
n	N	Kinematic hardening parameter
iplot	*flag*	Plot database flag: EQ. 1.0: Hardening variable EQ. 2.0: Cap - J_1 axis intercept $X_{(\kappa)}$ EQ. 3.0: Volumetric plastic strain ε^p_v EQ. 4.0: First stress invariant J_1 EQ. 5.0: Second stress invariant $\sqrt{J_{2D}}$ EQ. 6.0: Not used EQ. 7.0: Not used EQ. 8.0: Response mode number EQ. 9.0: Number of iterations
itype	*flag*	Formulation flag: EQ. 1.0: Soil or concrete (cap surface may contact) EQ. 2.0: Rock (cap surface may not contact)
ivec	*flag*	Vectorization flag: EQ. 0.0: Vectorized (fixed number of iterations) EQ. 1.0: Fully iterative
t	T	Tension cutoff, $T < 0$ (positive in compression)

The cap model is formulated in terms of the invariants of the stress tensor. The square root of the second invariant of the deviatoric stress tensor, $\sqrt{J_{2D}}$, is found from the deviatoric stresses $\boldsymbol{s}$ as

$$\sqrt{J_{2D}} \equiv \sqrt{\frac{1}{2} s_{ij} s_{ij}}. \tag{4-85}$$

The first invariant of the stress, J_1, is simply the sum of the normal stresses, or equivalently, three times the pressure.

The cap model consists of three surfaces in $\sqrt{J_{2D}}$ - J_1 space, as shown in Figure 4.9. First, there is a failure envelope surface, denoted f_1 in the figure. The functional form of f_1 is

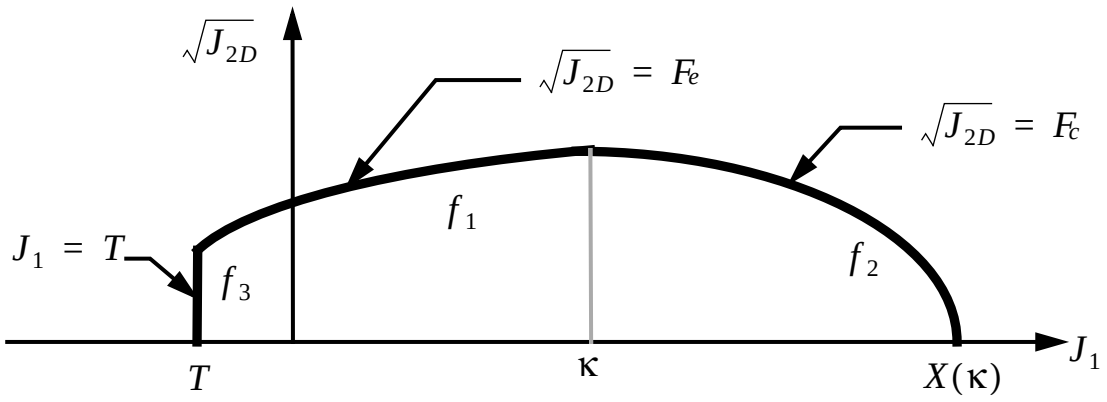


Figure 4.9
The yield surfaces of the two invariant cap model in $\sqrt{J_{2D}}$ - J_1 space. Surface f_1 is the failure envelope, f_2 is the cap surface, and f_3 is the tension cutoff.

$$f_1 = \sqrt{J_{2D}} - min(F_e(J_1), T_{mises})) \,, \tag{4-86}$$

where F_e is given by

$$F_e(J_1) \equiv \alpha - \gamma exp(-\beta J_1) + \theta J_1 \tag{4-87}$$

and $T_{mises} \equiv |X(\kappa_n) - L(\kappa_n)|$. This failure envelope surface is fixed in $\sqrt{J_{2D}}$ - J_1 space, and therefore does not harden, unless kinematic hardening is present. Next, there is a cap surface, denoted f_2 in the figure, with f_2 given by

$$f_2 = \sqrt{J_{2D}} - F_c(J_1, \kappa), \tag{4-88}$$

where F_c is defined by

$$F_c(J_1, \kappa) \equiv \frac{1}{R}\sqrt{[X(\kappa) - L(\kappa)]^2 - [J_1 - L(\kappa)]^2} \quad , \tag{4-89}$$

$X(\kappa)$ is the intersection of the cap surface with the J_1 axis,

$$X(\kappa) = \kappa + RF_e(\kappa) \ , \tag{4-90}$$

and $L(\kappa)$ is defined by

$$L(\kappa) \equiv \begin{cases} \kappa \ \text{ if } \ \kappa > 0 \\ 0 \ \text{ if } \ \kappa \le 0 \end{cases} . \tag{4-91}$$

The hardening parameter κ is related to the plastic volume change ε_v^p through the hardening law

$$\varepsilon_v^p = W\{1 - exp[-D(X(\kappa) - X_0)]\} \quad . \tag{4-92}$$

Geometrically, κ is seen in the figure as the J_1 coordinate of the intersection of the cap surface and the failure surface. Finally, there is the tension cutoff surface, denoted f_3 in the figure. The function f_3 is given by

$$f_3 \equiv T - J_1, \tag{4-93}$$

where T is an input material parameter which specifies the maximum hydrostatic tension sustainable by the material. The elastic domain in $\sqrt{J_{2D}}$ - J_1 space is then bounded by the failure envelope surface above, the tension cutoff surface on the left, and the cap surface on the right.

An additive decomposition of the strain into elastic and plastic parts is assumed:

$$\varepsilon = \varepsilon^e + \varepsilon^p, \tag{4-94}$$

where ε^e is the elastic strain and ε^p is the plastic strain. Stress is found from the elastic strain using Hooke's law,

$$\sigma = \boldsymbol{C}(\varepsilon - \varepsilon^p), \tag{4-95}$$

where σ is the stress and $\boldsymbol{C}$ is the elastic constitutive tensor.

The yield condition may be written: $f_1(\sigma) \le 0$; $f_2(\sigma) \le 0$; and $f_3(\sigma) \le 0$. The plastic consistency condition requires that

$$\begin{matrix} \dot{\lambda}_k f_k = 0 \\ \dot{\lambda}_k \ge 0 \end{matrix} \quad k = 1, 2, 3, \tag{4-96}$$

where λ_k is the plastic consistency parameter for surface k. If $f_k < 0$, then $\dot{\lambda}_k = 0$ and the response is elastic. If $f_k > 0$, then surface k is active and $\dot{\lambda}_k$ is found from the requirement that $\dot{f}_k = 0$.

Associated plastic flow is assumed;

$$\dot{\boldsymbol{e}}^p = \sum_{k=1}^{3} \dot{\lambda}_k \frac{\partial f_k}{\partial \sigma}. \tag{4-97}$$

Translation of the yield surfaces is permitted through the introduction of a "back stress" tensor, $\boldsymbol{a}$. The formulation including kinematic hardening is obtained by replacing the stress $\boldsymbol{s}$ with the translated stress tensor $\eta \equiv \sigma - \alpha$ in all of the above equations. The history tensor $\boldsymbol{a}$ is assumed deviatoric, and therefore has only 5 unique components. The evolution of the back stress tensor is governed by the nonlinear hardening law

$$\alpha = \bar{c}\bar{F}(\sigma, \boldsymbol{a})\dot{\boldsymbol{e}}^p , \tag{4-98}$$

where $\bar{c}$ is a constant, $\bar{F}$ is a scalar function of σ and α, and $\dot{\boldsymbol{e}}^p$ is the rate of deviatoric plastic strain. The constant $\bar{c}$ may be estimated from the slope of the shear stress - plastic shear strain curve at low levels of shear stress.

The function $\bar{F}$ is defined as

$$\bar{F} \equiv max\left(0, 1 - \frac{(\sigma - \alpha) \bullet \alpha}{2NF_e(J_1)}\right) , \tag{4-99}$$

where N is a constant defining the size of the yield surface.

The cap model contains a number of parameters which must be chosen to represent a particular material, and are generally based on experimental data. The parameters α, β, θ, and γ are usually evaluated by fitting a curve through failure data taken from a set of triaxial compression tests. The parameters W, D, and X_0 define the cap hardening law. The value of W represents the void fraction of the uncompressed sample and D governs the slope of the initial loading curve in hydrostatic compression. The value of R is the ratio of major to minor axes of the quarter ellipse defining the cap surface. Additional details may be found in (Chen and Baladi, 1985).

5.19 DYNA2D Material Type 19: Frazer-Nash Hyperelastic Rubber

Command	Variable	Description
ilimit	*ilimit*	Strain limit flag: EQ. 0.0: Stop if strain limits are exceeded EQ. 1.0: Continue if strain limits are exceeded
emax	E_{max}	Maximum strain limit
emin	E_{min}	Minimum strain limit
c100 .. c400	$C_{100} \ldots C_{400}$	Strain Energy Density Coefficients
c010 c020	$C_{010}\, C_{020}$	Strain Energy Density Coefficients
c110 c210	$C_{110}\, C_{210}$	Strain Energy Density Coefficients
c001 c101	$C_{001}\, C_{101}$	Strain Energy Density Coefficients

This model is a hyperelastic constitutive law representing the behavior of rubber-like materials at moderate to large strains. The strain energy density function is of the form

$$\begin{aligned} W \;=\; & C_{100}I_1 + C_{200}I_1^2 + C_{300}I_1^3 + C_{400}I_1^4 \\ & + C_{010}I_2 + C_{020}I_2^2 + C_{110}I_1I_2 + C_{210}I_1^2I_2 \\ & + C_{001}I_3 + C_{101}I_1I_3 \end{aligned} \tag{4-100}$$

where I_1, I_2, and I_3 are the strain invariants defined in terms of engineering components of the Green-Lagrange strain tensor $\boldsymbol{E}$ by

$$I_1 \;=\; E_{11} + E_{22} + E_{33}, \tag{4-101}$$

$$I_2 \;=\; (E_{11}E_{22} + E_{11}E_{33} + E_{22}E_{33}) - \frac{1}{4}(E_{12}^2 + E_{23}^2 + E_{31}^2)\,, \tag{4-102}$$

and

$$I_3 \;=\; \left(E_{11}E_{22}E_{33} + \frac{1}{4}E_{12}E_{23}E_{31}\right) - \frac{1}{4}(E_{11}E_{23}^2 + E_{22}E_{31}^2 + E_{33}E_{12}^2)\,. \tag{4-103}$$

The second Piola-Kirchhoff stress τ is found by differentiating the strain energy density function W with respect to the Green-Lagrange strain,

$$\tau \;=\; \frac{\partial W}{\partial \boldsymbol{E}}. \tag{4-104}$$

Cauchy stress σ is then found from the second Piola-Kirchhoff stress using

$$\sigma = \frac{1}{J}\boldsymbol{F}\tau\boldsymbol{F}^{T}, \tag{4-105}$$

where $\boldsymbol{F}$ is the deformation gradient and J is its determinant.

The model input includes a maximum strain limit E_{max}, a minimum strain limit E_{min}, and a strain limit option flag, $ILIMIT$. If the maximum normal strain is greater than E_{max} or the minimum normal strain is less than E_{min}, then a message is written to the screen and hsp printout file, and execution terminates if $ILIMIT = 0$ or continues if $ILIMIT = 1$.

5.20 DYNA2D Material Type 20: Laminated Composite

Command	Variable	Description
ea	E_a	Elastic modulus - fiber direction
eb	E_b	Elastic modulus - transverse direction
ec	E_c	Elastic modulus - thickness direction
vab	v_{ab}	Poisson's ratio, ab
vac	v_{ac}	Poisson's ratio, ac
vbc	v_{bc}	Poisson's ratio, bc
gab	G_{ab}	Shear modulus, ab
gac	G_{ac}	Shear modulus, ac
gbc	G_{bc}	Shear modulus, bc
t	*thickness*	Lamina thickness
aopt	A_{option}	Lamina axes option: EQ. 1.0: Planar laminate defined by normal *n* and interior surface point *P* EQ. 2.0: Spherical laminate with interior radius *R* and center *P*
n	*numlay*	Number of laminae
yp or rp	y_p or r_p	y-coordinate (or r-coordinate) of interior surface point *P*
zp	z_p	z-coordinate of interior surface point *P*
yn or rn	y_n or r_n	y-coordinate (or r-coordinate) of normal *n*
an	z_n	z-coordinate of normal *n*
R	*R*	Interior radius *R*
angle	$\Theta_1 ... \Theta_{numlay}$	lamina orientation angles (degrees)

Define $NUMLAY$ lamina orientation angles, θ, (in degrees) until all $NUMLAY$ angles have been defined.

This model simulates an elastic laminated composite by calculating the effective laminate properties, $\bar{\boldsymbol{C}}$, based upon the lamina present in each element. The effective laminate properties are calculated using the first order approach.

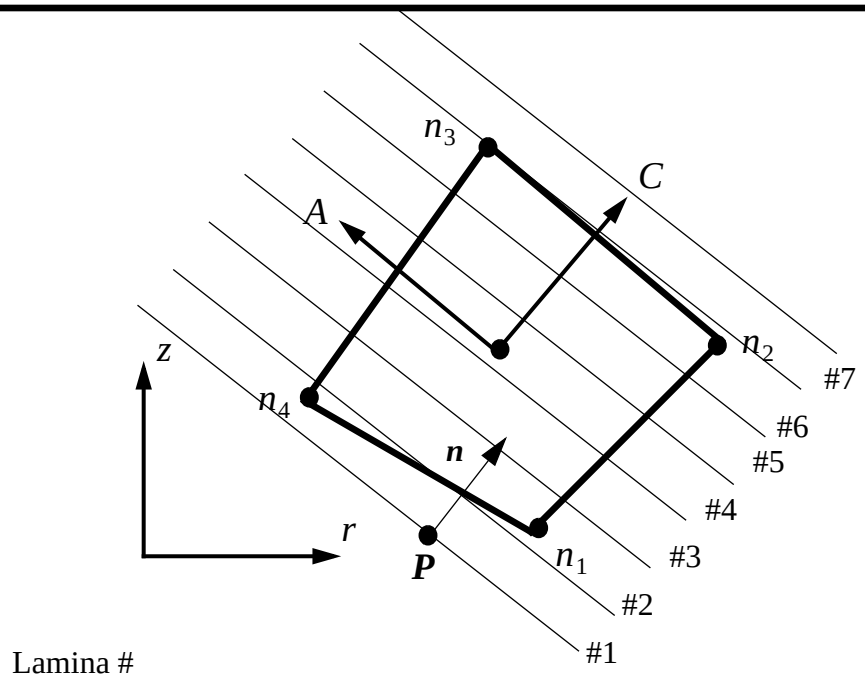
The material properties are specified in the lamina coordinate system **a**-**b**-**c**, where **a**, **b**, and **c** are the fiber, transverse, and through-thickness directions, respectively. The lamina orientation angle, θ, is the angle that the fibers make with the plane of the mesh, i.e. the $r-z$ or $y-z$ plane. For example, in axisymmetric geometries hoop lamina have $\theta = \pm 90^o$ while an axial lamina has $\theta = 0$ or $\theta = 180^o$.
The laminate geometry is based upon the lamination option and the assumption that each lamina is t thick. In the planar option, the i-th lamina ($i \neq 1$) exists in the region that is bound by lines that are perpendicular to **n** and pass through the points $\boldsymbol{P} + (i-1)t \times \boldsymbol{n}$ and $\boldsymbol{P} + i \times t \times \boldsymbol{n}$. In the spherical option, the i-th lamina ($i \neq 1$) exists in the region bound by two circles whose centers are **P** and radii are $R + (i-1) \times t$ and $R + i \times t$, respectively.

$$C_L^{-1} = \begin{bmatrix} \frac{1}{E_a} & -\frac{\nu_{ba}}{E_b} & -\frac{\nu_{ca}}{E_c} & 0 & 0 & 0 \\ -\frac{\nu_{ab}}{E_a} & \frac{1}{E_b} & -\frac{\nu_{cb}}{E_c} & 0 & 0 & 0 \\ -\frac{\nu_{ac}}{E_a} & -\frac{\nu_{bc}}{E_b} & \frac{1}{E_c} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G_{ab}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G_{bc}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G_{ca}} \end{bmatrix} . \qquad \text{(4-106)}$$

(4-107)

Poisson's ratios are defined as $\nu_{ij} = \frac{-\varepsilon_j}{\varepsilon_i}$

which represents the strain ratio resulting from a uniaxial stress applied in the *i-th* direction.



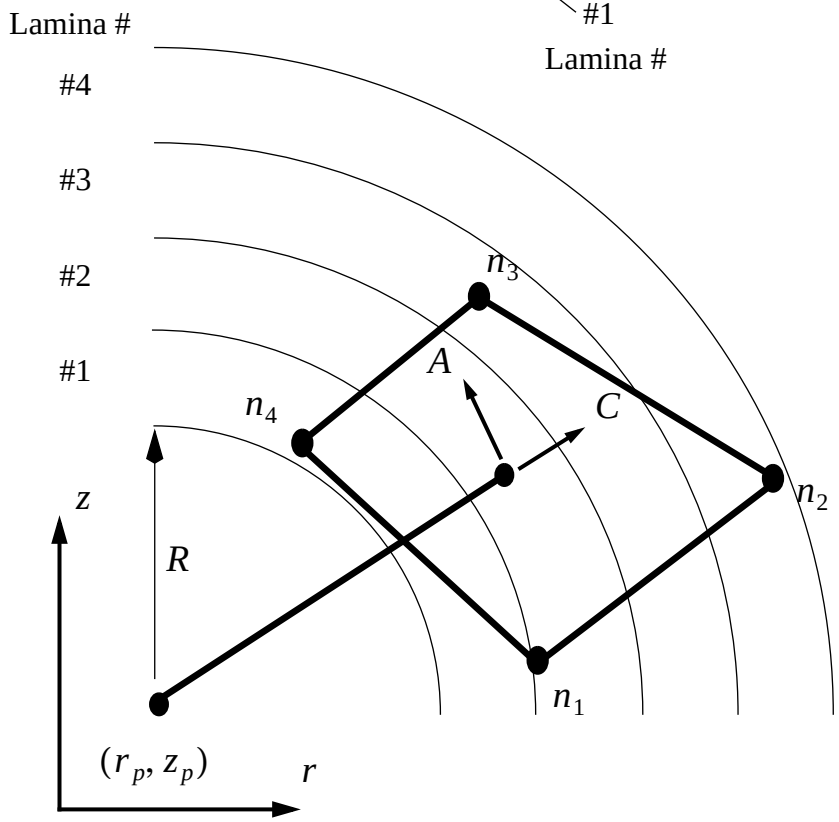


Figure 4.10
Options for determining the lamination axes
(a) AOPT = 1.0; (b) AOPT = 2.0.

Caution should be exercised when interpreting results. The two-dimensional assumptions used to reduce the full 6×6 stiffness matrix, $\bar{\boldsymbol{C}}$, to a 4×4 stiffness matrix are that the elemental shear stresses σ_{AB} and σ_{BC}, and strains γ_{AB} and γ_{BC} equal zero. For symmertic layups, this conditions is satisfied trivially since no off diagonal stiffness terms exit in $\bar{\boldsymbol{C}}$ to couple the AB and BC components to each other or to any other components. However, for nonsymmetric layups, non-zero off diagonal terms render it impossible to satisfying all four assumptions simultaneously. This material model is suggested for use only with symmetric layups.

5.21 DYNA2D Material Type 21: Isotropic-Elastic-Plastic

Command	Variable	Description
e	E	Young's modulus
v	ν	Poisson's ratio
sig0	σ_0	Yield stress
etan	E_T	Tangent modulus

This model produces bilinear elastoplastic behavior which is identical to Material Type 3 with $\beta = 1.0$, but is somewhat faster and requires less storage. The theoretical foundations of this model are similar to those described for Material Type 3. The numerical algorithms are based on those described in (Krieg and Key, 1976).

5.22 DYNA2D Material Type 22: Strain Rate Dependent Steinberg-Guinan-Lund

An equation of state must be used with this model.

Command	Variable	Description
g0	G_0	Shear modulus constant
sig0	σ_0	Yield stress constant
beta	β	Strain hardening law constant
n	n	Strain hardening exponent
gama	γ_i	Initial plastic strain
c1	C_1	First thermal activation constant
2uk	$2U_k$	Twice the dislocation kink energy
sigm	σ_m	Yield stress work hardening limit
b	b	Shear modulus pressure constant
bp	b'	Yield stress pressure constant
h	h	Energy coefficient
f	f	Energy exponential coefficient
yp	Y_p	Peierl's stress
c2	C_2	Second thermal activation constant
siga	σ_{am}	Maximum athermal yield stress
a	a	Atomic weight; if a = 0, R' must be defined
t0	T_{mo}	Melting temperature constant
gam0	γ_0	Thermodynamic gamma
sa	α	Thermodynamic constant
pc	P_{cut}	Pressure cutoff
trm	T_{room}	Room temperature
deby	Θ	Debye coefficient
spall	*spall*	Spall model flag: EQ. 0.0: default set to 2.0 EQ. 1.0: Pressure limit model EQ. 2.0: Maximum principal stress spall criterion EQ. 3.0: Hydrostatic tension spall criterion

rp	R'	NOTE: If $R' \neq 0$, atomic weight is not used
epsf	ε^{-p}_f	Effective plastic strain at failure
nfit	*nfit*	Polynomial order for fit; $1 \leq$ nfit ≤ 9
ivar	*flag*	Cold compression energy polynomial flag: EQ. 0.0: Polynomial coefficients given or fit in terms of η EQ. 1.0: Polynomial coefficients given or fit in terms of μ
min	*min_limit*	Optional minimum limit for energy fit. Input η_{min} if ivar = 0 Input μ_{min} if ivar = 0
max	*max_limit*	Optional maximum limit for energy fit. Input η_{max} if ivar = 0 Input μ_{max} if ivar = 0
ec0 ... ec9	$EC_0 \ldots EC_9$	Cold compression polynomial coefficients

The formulation of this model is essentially consists a strain rate modification of the Steinberg-Guinan model (Material Type 11) to extend the range of validity down to lower strain rates.

In terms of the foregoing input parameters, we define the shear modulus, G, before the material melts as:

$$G = G_0\left[1 + bpV^{\frac{1}{3}} - h\left(\frac{E_i - E_c}{3R'} - 300\right)\right]e^{-\frac{fE_i}{E_m - E_i}} \tag{4-108}$$

where p is the pressure, V is the relative volume, E_i is the current energy, E_c is the cold compression energy, and E_m is the melting energy. The cold compression energy is calculated using

$$E_c(x) = \int_0^x p\,dx\,, \tag{4-109}$$

where $x = 1 - V$. The equation is integrated using initial energy E_o and pressure P_o conditions that correspond to zero K and are given by

$$E_o = -3R'T_{room} \tag{4-110}$$

and

$$P_o = \gamma_o E_o \text{Debye}\left(\frac{\theta}{T_{room}}\right). \tag{4-111}$$

Here Debye is the Debye correction factor, and has a default value of 1 when $\theta = 0$. The melting energy is found from the cold compression energy and the melting temperature using

$$E_m(x) = E_c(x) + 3R'T_m(x) \ , \tag{4-112}$$

where the melting temperature T_m is given by

$$T_m(x) = \frac{T_{mo} exp(2ax)}{V^{2\left(\gamma_0 - a - \frac{1}{3}\right)}} \tag{4-113}$$

and T_{mo} is the melting temperature at the initial density, ρ_0.

In the above equations, R' is defined by

$$R' = \frac{R\rho}{A}, \tag{4-114}$$

where R is the universal gas constant and A is the atomic weight. **Note that if R' is not defined, DYNA2D computes it with R in the cm-gram-microsecond system of units.** Thus, this option should not be used unless the entire analysis model is defined in the cm-gram-microsecond second system of units.

The yield strength σ_y is decomposed into a thermally activated part σ_{yT} and an athermal part σ_{ya},

$$\sigma_y = \sigma_{yT}(\dot{\varepsilon}^p, T)G(P, T) + \sigma_{ya} \ , \tag{4-115}$$

where G is a dimensionless function that relates the shear modulus at the current pressure P and temperature T to the shear modulus under standard conditions.
The effective plastic strain rate is written in terms of the thermally activated yield stress and known functions as

$$\dot{\varepsilon}^p = \left\{ \frac{1}{C_1} exp\left[\frac{2U_k}{kT}\left(1 - \frac{\sigma_{yT}}{Y_P}\right)^2 \right] + \frac{C_2}{\sigma_{yT}} \right\}^{-1} \ , \tag{4-116}$$

where Y_P is the Peierls stress, $2U_k$ is the energy to form a pair of kinks in a dislocation segment of length L, and k is the Boltzmann constant. The constant C_2 is the drag coefficient D divided by the dislocation density ρ_d times the square of the Burger's vector b. The constant C_1 is given by

$$C_1 = \frac{\rho_d L a b^2 \nu}{2w^2}, \tag{4-117}$$

where a is the distance between Peierls valleys, w is the width of a kink loop, and ν is the Debye frequency. Equation (4-116) is solved iteratively to find σ_{YT}, but the additional restriction

$$\sigma_{YT} \le Y_P \tag{4-118}$$

is also imposed.

If E_m exceeds E_i (i.e., the material has not melted), then the athermal part of the yield strength σ_{ya} is given by:

$$\sigma_{ya} = \sigma_0' \left[1 + b' p V^{\frac{1}{3}} - h\left(\frac{E_i - E_c}{3R'} - 300 \right) \right] e^{-\frac{f E_i}{E_m - E_i}} . \tag{4-119}$$

The work-hardened yield stress σ'_0 is found from the initial yield stress σ_0 and the accumulated effective plastic strain $\bar{\varepsilon}^p$ using the hardening law

$$\sigma'_0 = \sigma_0 [1 + \beta(\gamma_i + \bar{\varepsilon}^p)]^n , \tag{4-120}$$

where γ_i is the initial plastic strain. If the work-hardened yield stress σ_0 exceeds the limiting value σ_m, then σ'_0 is reset to σ_m. After the materials melts ($E_i > E_m$), the athermal yield stress σ_{ya} and shear modulus G are reset to one half their initial value.

The evaluation of the cold compression energy $E_c(x)$ approximated with a polynomial during execution. The independent variable is chosen as η, and the polynomial takes the form

$$E_c = \sum_{i=0}^{NFIT-1} EC_i \eta^i \tag{4-121}$$

where $NFIT$ is the chosen order of the polynomial fit.
Note that the density and compression variables are related by

$$x = 1 - V = \frac{\mu}{\mu + 1} = 1 - \frac{1}{\eta} . \tag{4-122}$$

If the coefficients EC_0 through EC_9 are not specified in the input, DYNA2D will fit the cold compression energy with up to a ten term polynomial expansion using a least squares method. If the order of the polynomial is not specified, DYNA2D will automatically pick the best polynomial order that fits the EOS generated data. Otherwise, DYNA2D will attempt to fit the data to the polynomial order desired.

A Debye correction can be applied to the cold compression energy to improve the model's temperature response. This option is activated by specifying anon-zero value of the Debye coefficient θ.

A choice of three spall models is offered to represent material splitting, cracking, and failure under tensile loads. The pressure limit model, *SPALL = 1*, limits the hydrostatic tension to the specified value, p_{cut}. If pressures more tensile than this limit are calculated, the pressure is reset to p_{cut}. This option is not strictly a spall model, since the deviatoric stresses are unaffected by the pressure reaching the tensile cutoff, and the pressure cutoff value p_{cut} remains unchanged throughout the analysis. The maximum principal stress spall model, *SPALL = 2*, detects spall if the maximum (most tensile) principal stress σ_{max} exceeds the limiting value $-p_{cut}$. Note that the negative sign is required because p_{cut} is measured positive in compression, while σ_{max} is positive in tension. Once spall is detected with this model, the deviatoric stresses are set to zero, and no hydrostatic tension $(p < 0)$ is permitted. If tensile pressures are calculated, they are reset to 0 in the spalled material. Thus, the spalled material behaves as a rubble or incohesive material. The hydrostatic tension spall model, *SPALL = 3*, detects spall if the pressure becomes more tensile than the specified limit, p_{cut}. Once spall is detected the deviatoric stresses are set to zero, and the pressure is required to be compressive. If hydrostatic tension $(p < 0)$ is subsequently calculated, the pressure is reset to 0 for that element.

Material ‘‘erosion’’ and failure may be obtained by defining a nonzero effective plastic strain at failure $\bar{\varepsilon}_f^p$ and specifying that this material is active for automatic contact.

5.23 DYNA2D Material Type 23: Three-Invariant Viscoplastic Cap Model

Command	Variable	Description
g	G	Shear modulus
k	K	Bulk modulus
gr	Γ	Gruneisen ratio (optional)
pres	*flag*	Pore compression flag: EQ. 0.0: Explicit pore compression EQ. 1.0: Constant bulk modulus
alpha	α	Shear failure surface constant
theta	Θ	Shear failure surface linear coefficient
gamma	γ	Shear failure surface exponential coefficient
beta	β	Shear failure surface exponent
fail	t_{fail}	Tensile pressure cutoff (negative in tension)
mode		Tensile return mapping mode
nalpha	N^{α}	Kinematic hardening parameter
calpha	c^{α}	Kinematic hardening coefficient
r0	R_0	Initial ellipticity
x0	X_0	Initial J_1-axis intercept
rock	*option*	Cap contraction option: EQ. 0.0: Contraction allowed (soils) EQ. 1.0: Contraction omitted (rocks) EQ. 2.0: Contraction with hardening (rocks)
secp		Shear-enhanced compaction parameter
w	W	Maximum plastic volume strain
d1	D_1	Exponent D_1
d2	D_2	Exponent D_2
plot	*nplot*	Plot variable output option (see following table)
msi		Maximum strain increment
q1	Q_1	Three-invariant parameter
q2	Q_2	Three-invariant parameter: GE. 0.0: Formulation parameter LE. 0.0: Friction angle, φ(degrees)

db0	$\Delta\beta_0$	Rounded vertices parameter
delta	δ	Rounded vertices parameter
vfp		Viscoplasticity fluidity parameter
form	*form*	Viscoplastic flow function form: LT. 0.0: $\phi(f) = (f/f_0)^N$, where N = *form* GT. 0.0: $\phi(f) = exp(f/f_0)^N - 1$

Output variables for NPLOT plotting option.

NPLOT	Function	Description
1	$L(\kappa)$	J_1 value at cap-shear surface intersection
2	$X(\kappa)$	J_1 intercept of cap surface
3	$R(\kappa)$	Cap surface ellipticity
4	$\bar{\varepsilon}_v^p$	Plastic volume strain
5	J_1	First stress invariant
6	J_2'	Second invariant of deviatoric stress
7	J_3'	Third invariant of deviatoric stress
8	$\hat{J}_3'$	$(3\sqrt{1.5}J_3')/(J_2')^{3/2} = -\sin(3\beta)$
9	β	Lode angle (degrees)
10	R	Octahedral plane radius
11	J	Relative volume
12	ϕ	Porosity
13	ϕ_{cs}	Relative change in volume of solid phase
14	P_{hs}	Pressure in the solid phase
15	E_{hs}	Energy in the solid phase
16	nsubs	Number of strain subincrements
17	$1-(R^2F_f^2F_c)/J_2'$	Deviation from failure surface
18	G^α	Kinematic hardening limiting function
19*	J_2^α	Kinematic hardening backstress

5.24 DYNA2D Material Type 24: Bammann Plasticity Model

Command	Variable	Description
e	E	Young's modulus
v	ν	Poisson's ratio
t	T_0	Initial temperature (˚K)
hc	HC	Heat generation coefficient
axx	α_{xx}	Initial tensor internal variable α_{xx}
ayy	α_{yy}	Initial tensor internal variable α_{yy}
axy	α_{xy}	Initial tensor internal variable α_{xy}
k0	κ_0	Initial scale internal variable κ_0
epsf	$\bar{\varepsilon}_f^p$	Effective plastic strain at failure
c1 ... c18	$C_1 \ldots C_{18}$	

The number of material parameters may seem prohibitive, but rarely are all of the constants used. The model reduces to linear strain hardening with only two required parameters. For rate insensitive materials the number of parameters is reduced by four. If temperature dependence is not required (i.e. when heat generation is not important), then the number of parameters is reduced by a factor of two. All of the parameters can be determined using simple tension and compression data.

The evolution of the Cauchy stress σ is governed by an equation of the form

$$\dot{\sigma} = \lambda Tr(\boldsymbol{d}^e)\boldsymbol{1} + 2G\boldsymbol{d}^e, \tag{4-123}$$

where $\boldsymbol{d}^e$ is the elastic part of the rate of deformation, λ is the elastic Lame parameter given by

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}, \tag{4-124}$$

and G is the elastic shear modulus. The rate of deformation $\boldsymbol{d}$ (symmetric part of the velocity gradient) is decomposed as

$$\boldsymbol{d} = \boldsymbol{d}^e + \boldsymbol{d}^p + \boldsymbol{d}^{th}, \tag{4-125}$$

where $\boldsymbol{d}^p$ is the deviatoric plastic part, and $\boldsymbol{d}^{th}$ is the thermal expansion part. The deviatoric plastic part of the rate of deformation is given by

$$\boldsymbol{d}^p = f(T)\sinh\left[\frac{|\xi| - \kappa - Y(T)}{V(T)}\right]\frac{\xi}{|\xi|} \quad \text{for} \quad |\xi| - \kappa - Y(T) \geq 0 \tag{4-126}$$

and

$$\boldsymbol{d}^p = 0 \text{ for } (|\xi| - \kappa - Y(T)) < 0 \ , \tag{4-127}$$

where T is temperature, κ is a scalar hardening variable, ξ is the translated stress found from the deviatoric Cauchy stress σ and the tensor hardening variable α as

$$\xi \equiv \sigma - \frac{2}{3}\alpha \ , \tag{4-128}$$

and $f(T)$, $Y(T)$, and $V(T)$ are scalar functions. Assuming isotropic thermal expansion with coefficient $\hat{a}$, the thermal part of the rate of deformation can be written

$$\boldsymbol{d}^{th} = \hat{a}\dot{T}\mathbf{1} . \tag{4-129}$$

The evolution of the internal plasticity variables α and κ is found from

$$\dot{\alpha} = h(T)\boldsymbol{d}^p - [r_d(T)\bar{d} + r_s(T)]\bar{\alpha}\alpha \tag{4-130}$$

$$\dot{\kappa} = H(T)|\boldsymbol{d}^p| - [R_d(T)\bar{d} + R_s(T)]\kappa^2 \ , \tag{4-131}$$

where $h(T)$ and $H(T)$ are hardening moduli (which may be functions of temperature),

$$\bar{d} = \sqrt{\frac{2}{3}}|\boldsymbol{d}^p|, \tag{4-132}$$

$$\bar{\alpha} = \sqrt{\frac{2}{3}}|\alpha| \ , \tag{4-133}$$

and $r_s(T)$, $R_s(T)$, $r_d(T)$, and $R_d(T)$ are scalar functions.

To compute temperature change, it is assumed that no heat is conducted out of an element and 90% of the plastic work is dissipated as heat, so it follows that

$$\dot{T} = \frac{0.9}{\rho c_v}(\sigma \bullet \boldsymbol{d}^p) \ , \tag{4-134}$$

where ρ is the material density and c_v is the specific heat. To include this effect the heat generation coefficient, HC, should be defined in the input:

$$HC = \frac{0.9}{\rho c_v}. \tag{4-135}$$

Nine functions are used to describe the inelastic response. They can be grouped into three classes: those associated with the initial yield stress, the hardening functions, and the recovery functions. The temperature dependence of the yield functions are given by

$$V(T) = C_1 exp(-C_2/T) \tag{4-136}$$

$$Y(T) = C_3 exp(C_4/T) \tag{4-137}$$

$$F(T) = C_5 exp(-C_6/T) \ . \tag{4-138}$$

The function $Y(T)$ describes the rate independent yield strength as a function of temperature. The function $F(T)$ determines the rate at which the material transitions from rate-insensitive to rate-dependent, and $V(T)$ describes the amount of rate dependence.

Two internal state variables are used to model hardening. The tensor variable α is used to describe the translation of the yield surface and the scalar variable κ is used to track growth of the yield surface. These two history variables evolve independently, and their evolution is characterized by a hardening contribution minus a recovery contribution. The hardening functions $h(T)$ and $H(T)$ are given by

$$h(T) = C_9 exp(C_{10}/T) \tag{4-139}$$

$$H(T) = C_{15} exp(C_{16}/T) \ . \tag{4-140}$$

Without recovery terms the model reduces to linear hardening with a tangent modulus of

$$E_T = (E(h + H))/(E + h + H) \ . \tag{4-141}$$

There are two recovery functions associated with each of the state variables α and κ. Larger values of recovery result in faster deviation from linear hardening and lower saturation stresses. The dynamic recovery function results in rate-independent hardening while the static (or thermal) recovery results in rate-dependent hardening.

The recovery functions are strongly temperature-dependent, and their form is given by

$$r_d(T) = C_7 exp(-C_8/T) \tag{4-142}$$

$$r_s(T) = C_{11} exp(-C_{12}/T) \tag{4-143}$$

$$R_d(T) = C_{13} exp(-C_{14}/T) \tag{4-144}$$

$$R_s(T) = C_{17} exp(-C_{18}/T) \ . \tag{4-145}$$

At higher strain rates and lower temperatures the dynamic recovery is dominant while at lower strain rates and higher temperatures the static recovery is dominant.

For high rate problems there can be a significant temperature increase due to plastic work. This allows the model to calculate thermal softening and thermal instabilities. Note that the heat generation coefficient HC will have no effect unless the functions are temperature-dependent. Typically, for strain rates less than 1.0 $\frac{1}{\text{sec}}$ the problem is not adiabatic and therefore the heat generation coefficient should *not* be included.

The parameters that give initial values to the components of the internal variable α may often be defined as zero. Nonzero values may be used to describe a material that is not initially isotropic, such as material deformed by a rolling process.

Material ‘‘erosion’’ and failure may be obtained by defining a nonzero effective plastic strain at failure $\bar{\varepsilon}_f^p$ and specifying that this material is active for automatic contact.

5.25 DYNA2D Material Type 25: Sandia Damage Model

Command	Variable	Description
e	E	Young's modulus
v	ν	Poisson's ratio
t	T_0	Initial temperature ($^\circ$K)
hc	HC	Heat generation coefficient
axx	α_{xx}	Initial tensor internal variable α_{xx}
ayy	α_{yy}	Initial tensor internal variable α_{yy}
axy	α_{xy}	Initial tensor internal variable α_{xy}
k0	κ_0	Initial scale internal variable κ_0
mbar	$\bar{m}$	Damage exponent
d0	D_0	Initial void volume fraction (porosity)
epsf	$\bar{\varepsilon}^p_f$	Effective plastic strain at failure
fsd	*level*	Element deletion controlled by damage level: EQ.0: Damage level does not control element deletion GT. 0: Element deletion based upon damage, $\dot{D}$
c1 ... c18	$C_1 \ldots C_{18}$	

This is a phenomenological plasticity model using a set of internal state variables whose evolution is based on micromechanics. The model includes rate and temperature dependence, and heat generation due to plastic work. Since internal state variables are used to track the deformation, the history effects of strain rate and temperature are correctly captured. Ductile failure in materials is predicted by the model using a void growth evolution law.

The evolution of the Cauchy stress σ is governed by an equation of the form

$$\dot{\sigma} = \lambda(1-D)Tr(\boldsymbol{d}^e)\boldsymbol{1} + 2G(1-D)\boldsymbol{d}^e - \frac{D}{1-D}\sigma \ , \tag{4-146}$$

where $\boldsymbol{d}^e$ is the elastic part of the rate of deformation, D is a scalar damage variable, λ is the elastic Lame parameter given by

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} \ , \tag{4-147}$$

and G is the elastic shear modulus. The rate of deformation $\boldsymbol{d}$ (symmetric part of the velocity gradient) is decomposed as

$$\boldsymbol{d} = \boldsymbol{d}^e + \boldsymbol{d}^p + \boldsymbol{d}^v + \boldsymbol{d}^{th}, \tag{4-148}$$

where $\boldsymbol{d}^p$ is the deviatoric plastic part, $\boldsymbol{d}^v$ is the dilatational plastic part, and $\boldsymbol{d}^{th}$ is the thermal expansion part. The deviatoric plastic part of the rate of deformation is given by

$$\boldsymbol{d}^p = f(T)\sinh\left[\frac{|\xi| - \kappa - Y(T)(1-D)}{V(T)(1-D)}\right]\frac{\xi}{|\xi|} \quad \text{for} \quad |\xi| - \kappa - Y(T)(1-D) \geq 0 \tag{4-149}$$

and

$$\boldsymbol{d}^p = 0 \text{ for } \quad (|\xi| - \kappa - Y(T)(1-D)) < 0 \quad , \tag{4-150}$$

where T is temperature, κ is a scalar hardening variable, ξ is the translated stress found from the deviatoric Cauchy stress σ and the tensor hardening variable α as

$$\xi \equiv \sigma - \frac{2}{3}\alpha, \tag{4-151}$$

and $f(T)$, $Y(T)$, and $V(T)$ are scalar functions. The diliational plastic part of the rate of deformation depends only on the damage variable and is given by

$$\boldsymbol{d}^v = \frac{D}{1-D}\mathbf{1}. \tag{4-152}$$

Assuming isotropic thermal expansion with coefficient $\hat{a}$, the thermal part of the rate of deformation can be written

$$\boldsymbol{d}^{th} = \hat{a}\dot{T}\mathbf{1}. \tag{4-153}$$

The evolution of the internal plasticity variables α and κ is found from

$$\dot{\alpha} = h(T)\boldsymbol{d}^p - [r_d(T)\bar{d} + r_s(T)]\bar{\alpha}\alpha \tag{4-154}$$

$$\dot{\kappa} = H(T)|\boldsymbol{d}^p| - [R_d(T)\bar{d} + R_s(T)]\kappa^2 \quad , \tag{4-155}$$

where $h(T)$ and $H(T)$ are hardening moduli (which may be functions of temperature),

$$\bar{d} = \sqrt{\frac{2}{3}}|\boldsymbol{d}^p|, \tag{4-156}$$

$$\bar{\alpha} = \sqrt{\frac{2}{3}}|\alpha| , \tag{4-157}$$

and $r_s(T)$, $R_s(T)$, $r_d(T)$, and $R_d(T)$ are scalar functions.

The evolution of the damage parameter D is given by

$$\dot{D} = \chi\left[\frac{1}{(1-D)^{\bar{m}}} - (1-D)\right]|\boldsymbol{d}^p| , \tag{4-158}$$

where χ is a stress triaxiality factor given by

$$\chi = \sinh\left[\frac{2(2\bar{m}-1)p}{(2\bar{m}+1)\bar{\sigma}}\right] , \tag{4-159}$$

$\bar{m}$ is a void growth constant, p is pressure, and $\bar{\sigma}$ is effective stress.

To compute temperature change, it is assumed that no heat is conducted out of an element and 90% of the plastic work is dissipated as heat, so it follows that

$$\dot{T} = \frac{0.9}{\rho c_v}(\sigma \bullet \boldsymbol{d}^p) , \tag{4-160}$$

where ρ is the material density and c_v is the specific heat. To include this effect the heat generation coefficient, HC, should be defined in the input:

$$HC = \frac{0.9}{\rho c_v} . \tag{4-161}$$

Nine functions are used to describe the inelastic response. They can be grouped into three classes: those associated with the initial yield stress, the hardening functions, and the recovery functions. The temperature dependence of the yield functions are given by

$$V(T) = C_1 exp(-C_2/T) \tag{4-162}$$

$$Y(T) = C_3 exp(C_4/T) \tag{4-163}$$

$$F(T) = C_5 exp(-C_6/T) . \tag{4-164}$$

The function $Y(T)$ describes the rate independent yield strength as a function of temperature. The function $F(T)$ determines the rate at which the material transitions from rate-insensitive to rate-dependent, and $V(T)$ describes the amount of rate dependence.

Two internal state variables are used to model hardening. A tensor variable α is used to describe the translation of the yield surface and a scalar variable κ is used to track growth of the yield surface. These two history variables evolve independently, and their evolution is characterized by a hardening contribution minus a recovery contribution. The hardening functions $h(T)$ and $H(T)$ are given by

$$h(T) = C_9 exp(C_{10}/T) \tag{4-165}$$

$$H(T) = C_{15} exp(C_{16}/T) \ . \tag{4-166}$$

Without recovery terms the model reduces to linear hardening with a tangent modulus of

$$E_T = (E(h + H))/(E + h + H) \ . \tag{4-167}$$

There are two recovery functions associated with each of the state variables α and κ. Larger values of recovery result in faster deviation from linear hardening and lower saturation stresses. The dynamic recovery function results in rate-independent hardening while the static (or thermal) recovery results in rate-dependent hardening.

The recovery functions are strongly temperature-dependent, and their form is given by

$$r_d(T) = C_7 exp(-C_8/T) \tag{4-168}$$

$$r_s(T) = C_{11} exp(-C_{12}/T) \tag{4-169}$$

$$R_d(T) = C_{13} exp(-C_{14}/T) \tag{4-170}$$

$$R_s(T) = C_{17} exp(-C_{18}/T) \ . \tag{4-171}$$

At higher strain rates and lower temperatures the dynamic recovery is dominant while at lower strain rates and higher temperatures the static recovery is dominant.

Material ‘‘erosion’’ and failure may be obtained by defining a nonzero effective plastic strain at failure $\bar{\varepsilon}_f^p$ and specifying that this material is active for automatic contact. Alternatively, by specifying a positive FSD and nonzero damage parameters, material ‘‘erosion’’ will be controlled by the accumulated damage parameter $\dot{D}$. Both material erosion methods can be used simultaneously. In this case which every condition is satisfied first will determine erosion.

5.26 DYNA2D Material Type 26: Circumferentially Cracked Elastoplasticity

Command	Variable	Description
e	E	Young’s modulus$_i$
v	ν	Poisson’s ratio
sigy	σ_o	Yield stress
ef	ε_f	Effective plastic strain at failure
etan	E_p	Hardening modulus$_i$
beta	β	Elastic modulus: $0 \le \beta \le 1$ coefficient
npts	n	Number of points in stress-effective plastic strain curve
eps	$\varepsilon_1 \ldots \varepsilon_n$	Effective plastic strain
es	$\sigma_1 \ldots \sigma_n$	Effective stress

This model is applicable only to axisymmetric structures. The material behavior is identical to Material Type 3, except that elements of this material cannot carry tensile circumferential stress. By specifying an initial relative volume greater than one on the element cards, the development of compressive circumferential stresses can be delayed or prevented since a gap must close circumferentially before compressive hoop stress develops.

6 DYNA2D EQUATIONS-OF-STATE

An equation-of-state defines the volumetric behavior of a material, and must be used *only* in combination with a hydrodynamic material model. Many equation-of-state models allow the user to specify an initial energy per unit initial volume, E_0, and an initial relative volume, V_0. If $V_0 \neq 0$, then E_0 is the energy per unit initial volume (i.e., per unit reference volume for the relative volume computation). Define an equation-of-state only for Material Types 8, 9, 10, 11, 12, 15, 16, and 22.

6.1 General Equation-of-State Definition Commands

The following commands will override the default falues for the current material only.

head *heading*	Equation-of-state heading. Command is entered in form: head *heading*.

6.2 Equation-of-State Form 1: Linear Polynomial

Command	Variable	Description
c0	C_0	Pressure constant
c1	C_1	Linear compression coefficient
c2	C_2	Quadratic compression coefficient
c3	C_3	Cubic compression coefficient
c4	C_4	First energy coefficient
c5	C_5	Second energy coefficient
c6	C_6	Third energy coefficient
e0	E_0	Initial internal energy per unit initial volume
v0	V_0	Initial relative volume

The linear polynomial equation-of-state is linear in internal energy. The pressure is given by:

$$p = C_0 + C_1\mu + C_2\bar{\mu}^2 + C_3\mu^3 + (C_4 + C_5\mu + C_6\bar{\mu}^2)E \quad (5\text{-}1)$$

where the excess compression μ is given by

$$\mu \equiv \frac{\rho}{\rho_0} - 1 , \quad (5\text{-}2)$$

E is the internal energy, ρ is the current density, and ρ_0 is the initial density. The tension-limited excess compression $\bar{\mu}$ is given by

$$\bar{\mu} = max(\mu, 0) . \tag{5-3}$$

Relative volume is related to excess compression and density by

$$V = \frac{1}{1+\mu} = \frac{\rho_0}{\rho} . \tag{5-4}$$

If $C_1 = K$ (the elastic bulk modulus) and all other $C_i = 0$, then linear elastic volumetric response is obtained.

6.3 Equation-of-State Form 2: JWL

Command	Variable	Description
a	A	
b	B	
r1	R_1	
r2	R_2	
omega	ω	
e0	E_8	Initial internal energy per unit initial volume
v0	V_0	Initial relative volume

The JWL equation of state is often used for detonation products of high explosives. The JWL equation-of-state defines the pressure as

$$p = A\left(1 - \frac{\omega}{R_1 V}\right)e^{-R_1 V} + B\left(1 - \frac{\omega}{R_2 V}\right)e^{-R_2 V} + \frac{\omega E}{V} , \tag{5-5}$$

where V is relative volume and E is internal energy.

6.4 Equation-of-State Form 3: Sack

Command	Variable	Description
e0	E_8	Initial internal energy per unit initial volume
v0	V_0	Initial relative volume
a1, a2, a3	A_1, A_2, A_3	
b1, b2	B_1, B_2	

This equation-of-state form is often used for detonation products of high explosives. The Sack equation-of-state defines pressure p as

$$p = \frac{A_3}{V^{A_1}} e^{-A_2 V}\left(1 - \frac{B_1}{V}\right) + \frac{B_2}{V} E \ , \tag{5-6}$$

where V is relative volume and E is the internal energy.

6.5 Equation-of-State Form 4: Gruneisen

Command	Variable	Description
sp	C	Velocity curve intercept
s1	S_1	First slope coefficient
s2	S_2	Second slope coefficient
s3	S_3	Third slope coefficient
gamma	γ_0	Gruneisen coefficient
sa	a	First order volume correction coefficient
e0	E_8	Initial internal energy per unit initial volume
v0	V_0	Initial relative volume

The Gruneisen equation-of-state with cubic shock velocity-particle velocity defines pressure for compressed materials ($\mu > 0$) as

$$p = \frac{\rho_0 C^2 \mu\left[1 + \left(1 - \frac{\gamma_0}{2}\right)\mu - \frac{a}{2}\mu^2\right]}{\left[1 - (S_1 - 1)\mu - S_2\frac{\mu^2}{\mu + 1} - S_3\frac{\mu^3}{(\mu + 1)^2}\right]^2} + (\gamma_0 + a\mu)E \quad , \tag{5-7}$$

and for expanded materials ($\mu < 0$) as

$$p = \rho_0 C^2 \mu + (\gamma_0 + a\mu)E, \tag{5-8}$$

where C is the intercept of the shock velocity vs. particle velocity ($v_s - v_p$) curve, S_1, S_2, and S_3 are the coefficients of the slope of the $v_s - v_p$ curve, γ_0 is the Gruneisen gamma, and a is the first order volume correction to γ_0. The excess compression μ is defined by

$$\mu \equiv \frac{\rho}{\rho_0} - 1, \tag{5-9}$$

where ρ is the current density and ρ_0 is the initial density.

6.6 Equation-of-State Form 5: Ratio of Polynomials

Command	Variable	Description
alpha	α	
beta	β	
e0	E_8	Initial internal energy per unit initial volume
v0	V_0	Initial relative volume
a10 ... a14	$A_{10} \ldots A_{14}$	
a20 ... a24	$A_{20} \ldots A_{24}$	
a30 ... a33	$A_{30} \ldots A_{33}$	
a40 ... a43	$A_{40} \ldots A_{43}$	
a50 ... a53	$A_{50} \ldots A_{53}$	
a60 ... a63	$A_{60} \ldots A_{63}$	
a70 ... a73	$A_{70} \ldots A_{73}$	

The ratio of polynomials equation-of-state defines the pressure as

$$p = \frac{F_1 + F_2 E + F_3 E^2 + F_4 E^3}{F_5 + F_6 E + F_7 E^2}(1 + \alpha\mu) \tag{5-10}$$

where the each of the F_i are polynomials in terms of the excess compression μ of the form

$$F_i = \sum_{k=0}^{n} A_{ik}\mu^k \tag{5-11}$$

with $n = 4$ for F_1 and F_2, and $n = 3$ for F_3 through F_7. The excess compression μ is defined by

$$\mu \equiv \frac{\rho}{\rho_0} - 1 , \tag{5-12}$$

where ρ is the current density and ρ_0 is the initial density. In expanded elements ($\mu < 0$), F_1 in (5-10) is replaced by $\bar{F}_1$, where $\bar{F}_1$ is defined by

$$\bar{F}_1 = F_1 + \beta\mu^2 . \tag{5-13}$$

6.7 Equation-of-State Form 6: Linear Polynomial with Energy Deposition

Command	Variable	Description
c0	C_0	Pressure constant
c1	C_1	Linear compression coefficient
c2	C_2	Quadratic compression coefficient
c3	C_3	Cubic compression coefficient
c4	C_4	First energy coefficient
c5	C_5	Second energy coefficient
c6	C_6	Third energy coefficient
e0	E_8	Initial internal energy per unit initial volume
v0	V_0	Initial relative volume
lc	*load_curve*	Load curve number giving energy deposition rate

This equation-of-state form is similar to equation-of-state form 1 except that this form allows internal energy to be deposited into the material at a specified rate. The pressure is given by:

$$p = C_0 + C_1\mu + C_2\bar{\mu}^2 + C_3\mu^3 + (C_4 + C_5\mu + C_6\bar{\mu}^2)E \tag{5-14}$$

where the excess compression μ is given by

$$\mu \equiv \frac{\rho}{\rho_0} - 1 , \tag{5-15}$$

E is the internal energy, ρ is the current density, and ρ_0 is the initial density. The tension-limited excess compression $\bar{\mu}$ is given by

$$\bar{\mu} = max(\mu, 0) . \tag{5-16}$$

Relative volume is related to excess compression and density by

$$V = \frac{1}{1+\mu} = \frac{\rho_0}{\rho}. \qquad (5\text{-}17)$$

Internal energy is added into the material at a rate specified by *load_curve*. This allows energy transfer mechanisms to be included which are not considered in detail in the analysis model.

This equation-of-state is linear in internal energy, and may be used to fit experimental data for many materials.

.

6.8 Equation-of-State Form 7: Ignition and Growth of Reaction in HE

Command	Variable	Description
fr	F_r	Second ignition coefficient
fmig	$F_{max.\ ignition}$	
fq	F_q	
g1	G_1	
m	m	
a1	a_1	
s1	s_1	
cvp	C_{vp}	Heat capacity of reaction products
cve	C_{ve}	Heat capacity of unreacted HE
ccrit	$C_{critical}$	
qr	Q_r	
t0	T_0	Initial temperature (˚K)
g2	G_2	
a2	a_2	
s2	s_2	
n	n	
fmgr	$F_{max.\ gr}$	
fmcp	$F_{min.\ gr}$	
r1p, r2p, r3p, r5p, r6p	$R_{1p,}\ R_{2p,}$ $R_{3p,}\ R_{5p,}\ R_{6p}$	
r1e r2e, r3e, r5e, r6e	$R_{1e,}\ R_{2e,}$ $R_{3e,}\ R_{5e,}\ R_{6e}$	

In the following description, a subscript "*e*" denotes quantities for the unreacted explosive, a subscript "p" denotes quantities for the reaction products, p is the pressure, V is the relative volume, and T is the absolute temperature.

The pressure in the unreacted explosive is given by a JWL equation of state,

$$p_e = R_{1e}e^{-R_{5e}V_e} + R_{2e}e^{-R_{6e}V_e} + R_{3e}\frac{T_e}{V_e} \tag{5-18}$$

where R_{3e} is a given constant related to the specific heat c_{ve} and JWL parameter ω by

$$R_{3e} = \omega_e c_{ve}. \tag{5-19}$$

The pressure in the reaction products is defined by another JWL equation of state,

$$p_p = R_{1p}e^{-R_{5p}} + R_{2p}e^{-R_{6p}V_p} + R_{3p}\frac{T_p}{V_p}, \tag{5-20}$$

and R_{3p} is a given constant related to the specific heat and JWL parameter as above.

As the chemical reaction converts unreacted explosive to reaction products, these JWL equations of state are used to calculate the pressure in the mixture. This mixture is defined by the fraction reacted, F, where $F = 0$ represents no reaction (all explosive) and $F = 1.0$ represents complete reaction (all products). The temperatures and pressures of reactants and products are assumed to be in equilibrium (i.e., $T_e = T_p$ and $p_e = p_p$), and the relative volumes are additive,

$$V = (1-F)V_e + V_p. \tag{5-21}$$

The assumed forms for the chemical reaction rate is then

$$\frac{\partial F}{\partial t} = \dot{F}_1 + \dot{F}_2 + \dot{F}_3, \tag{5-22}$$

where the ignition term is given by

$$\dot{F}_1 = F_q(1-F)^{F_r}\left[\frac{1}{V_e} - 1 - C_{crit}\right]^{\eta}, \tag{5-23}$$

the growth term is given by

$$\dot{F}_2 = G_1(1-F)^{s_1}F^{a_1}p^m, \tag{5-24}$$

and the completion term is given by

$$\dot{F}_3 = G_2(1-F)^{s_2}F^{a_2}p^n, \tag{5-25}$$

and F_q, F_r, C_{crit}, η, G_1, s_1, a_1, m, G_2, s_2, a_2, and n are given constants.

The ignition rate $\dot{F}_1$ is set to zero when $F \geq F_{max,ig}$, the growth term $\dot{F}_2$ is set to zero when $F \geq F_{max,gr}$, and the completion term $\dot{F}_3$ is set to zero when $F \leq F_{min,gr}$.

6.9 Equation-of-State Form 8: Tabulated with Compaction

Command	Variable	Description
npts	n	Number of points in tabulated curves
lnv	$\varepsilon_{v1} \ldots \varepsilon_{v10}$	
pc	$C_1 \ldots C_{10}$	
pt	$T_1 \ldots T_{10}$	
ku	$K_1 \ldots K_{10}$	
gamma	γ	
e0	E_8	Initial internal energy per unit initial volume
v0	V_0	Initial relative volume

Pressure is positive in compression, and volumetric strain ε_v is positive in tension. The tabulated compaction model is linear in internal energy. Pressure is defined by

$$p = C(\varepsilon_v) + \gamma T(\varepsilon_v)E \tag{5-26}$$

during loading (compression). Unloading occurs at a slope corresponding to the bulk modulus at the peak (most compressive) volumetric strain, as shown in Figure 5.1. Reloading follows the unloading path to the point where unloading began, and then continues on the loading path described by (5-26).

The volumetric strain is found from the relative volume V as

$$\varepsilon_v = ln(V)\,. \tag{5-27}$$

The tabulated functions may contain from 2 to 10 points, and the model will extrapolate using the last two points to find the pressure if required.

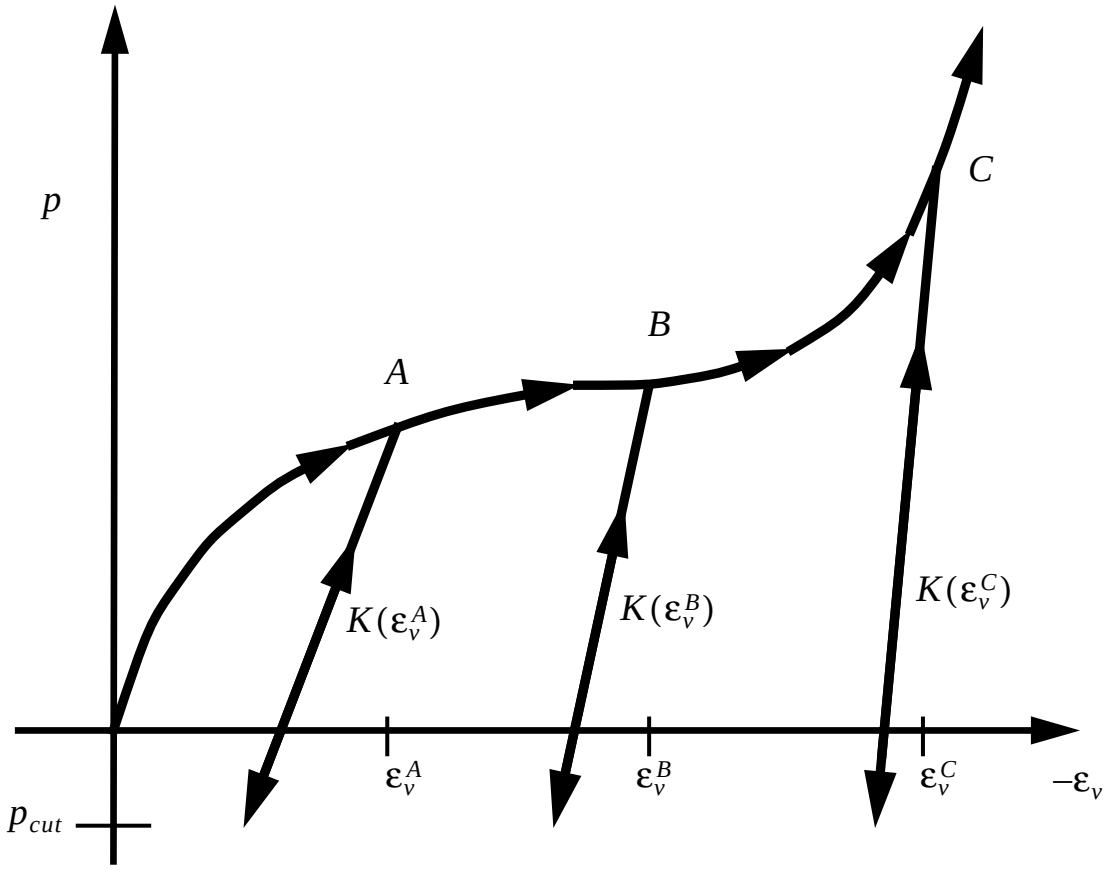


Figure 5.1

Pressure vs. volumetric strain curve for equation-of-state Form 8 with compaction. The unloading bulk modulus K depends on the peak volumetric strain. The tension limit p_{cut} is defined in the material model definition. Note p is positive in compression and ε_v is positive in tension.

6.10 Equation-of-State Form 9: Tabulated

Command	Variable	Description
ev1 ... ev10	$\varepsilon_{v1} \ldots \varepsilon_{v10}$	
c1 ... c10	$C_1 \ldots C_{10}$	
t1 ... t10	$T_1 \ldots T_{10}$	
gamma	γ	
e0	E_8	Initial internal energy per unit initial volume
v0	V_0	Initial relative volume

Pressure p is positive in compression, and volumetric strain ε_v is positive in tension. The tabulated compaction model is linear in internal energy. Pressure is defined by

$$p = C(\varepsilon_v) + \gamma T(\varepsilon_v) E \ , \tag{5-28}$$

where E is internal energy. The volumetric strain is found from the relative volume V as

$$\varepsilon_v = ln(V) \ . \tag{5-29}$$

Tabulated functions may contain from 2 to 10 points. The model will extrapolate to find the pressure if required.

6.11 Equation-of-State Form 10: Propellant

Command	Variable	Description
beta	β	Burn rate
alpha	α	Pressure exponent
theta	Θ	Form factor
gamma	γ	Ratio of specific heats
rho	ρ_s	Solid density
dweb	d_{web}	Web thickness
tf	T_f	Flame temperature
mc	M_c	Gas molecular weight
r	R	Gas constant
h	η	Specific covolume
fmi	f_{mi}	Igniter mass fraction

6.12 Equation-of-State Form 11: Pore Collapse

Command	Variable	Description
nld	*nld*	Number of virgin loading curve points
ncr	*ncr*	Number of completely crushed curve points
mu1	μ_1	Excess compression required before any pores can collapse
mu2	μ_2	Excess compression point where the virgin loading curve and the completely crushed curve intersect
e0	E_o	Initial internal energy per unit initial volume
mu0	μ_0	Internal excess compression
muvlc	μ	Virgin loading curve excess compression
puvlc	p	Virgin loading curve excess corresponding pressure
muccc	μ	Completely crushed curve excess compression
pccc	p	Completely crushed curve corresponding pressure

The Pore Collapse equation-of-state uses two curves: the virgin loading curve and the completely crushed curve, as shown in Figure 5.2. Two critical points are defined: the excess compression point required for pore collapse to begin (μ_1), and the excess compression point required to completely crush the material (μ_2). From this data and the maximum compression the material has attained (μ_{max}), the pressure for any excess compression μ can be determined. Unloading occurs along the virgin loading curve until the excess compression surpasses μ_1. After that, the unloading follows a path between the completely crushed curve and the virgin loading curve. Reloading will follow this curve back up to the virgin loading curve. Once the excess compression exceeds μ_2, then all unloading will follow the completely crushed curve.

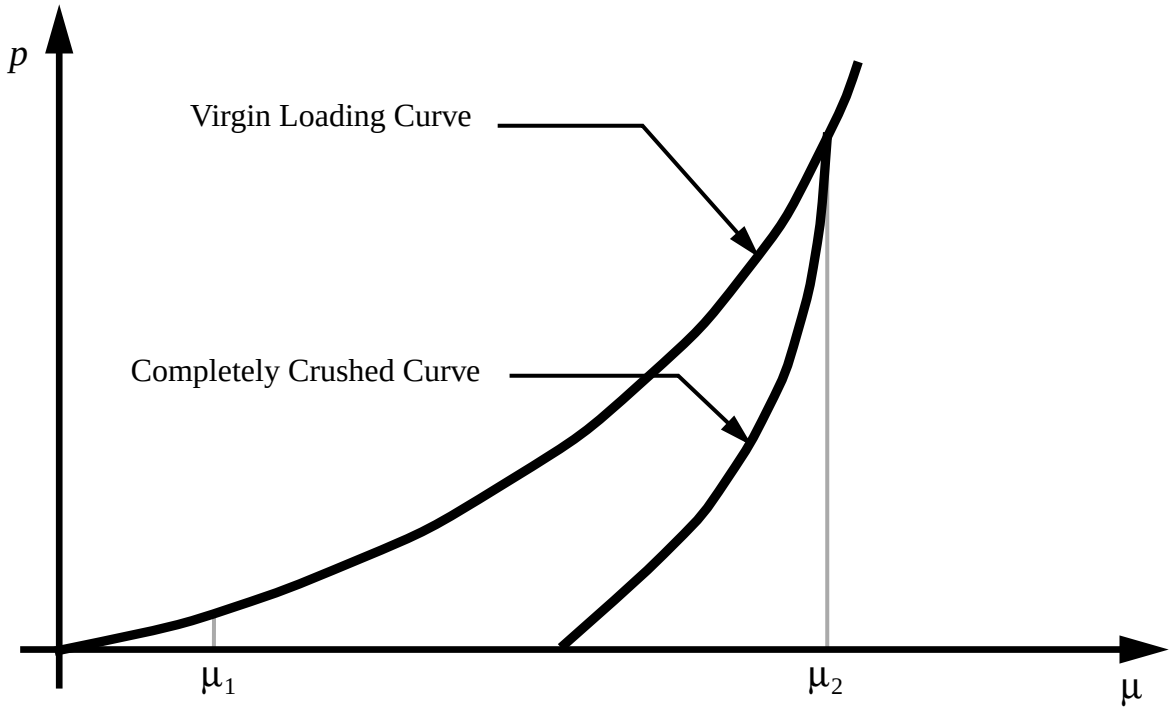


Figure 5.2
Pressure vs. excess compression for the Pore Collapse equation-of-state. The partially crushed curve lies between the Virgin Loading Curve and the Completely Crushed Curve.

For unloading between μ_1 and μ_2 a partially crushed curve is determined by the relationship:

$$p_{pc}(\mu) = p_{cc}\left(\frac{(1+\mu_B)(1+\mu)}{1+\mu_{max}} - 1\right) \quad , \tag{5-30}$$

where

$$\mu_B = p_{cc}^{-1}(p_{max}) \tag{5-31}$$

is the excess compression corresponding to a pressure of p_{max} on the completely crushed curve. In the above, subscript pc refers to the partially crushed state and subscript cc refers to the completely crushed state.

In terms of the relative volume V,

$$V = \frac{1}{1+\mu}, \tag{5-32}$$

$$p_{pc}(V) = p_{cc}\left(\frac{V_B}{V_{min}}V\right). \tag{5-33}$$

Thus, for a fixed $V_{min} = \frac{1}{\mu_{max}+1}$, the partially crushed curve will separate linearly from the completely crushed curve as V increases to account for pore recovery in the material.

The bulk modulus K is determined as one plus the excess compression times the slope of the current curve,

$$K = (1+\mu)\frac{\partial p}{\partial \mu}. \tag{5-34}$$

It then follows that the slope for the partially crushed curve is

$$\frac{\partial p}{\partial \mu} = \frac{(1+\mu_B)}{1+\mu_{max}} \frac{\partial p_{cc}}{\partial \mu}\left(\frac{(1+\mu_B)(1+\mu)}{(1+\mu_{max})}\right) \quad . \tag{5-35}$$

The bulk sound speed is determined from the slope of the completely crushed curve at the current pressure to avoid instabilities.

6.13 Equation-of-State Form 12: Composite HE

Command	Variable	Description
ap	A_p	JWL coefficient for component 1 products
bp	B_p	JWL coefficient for component 1 products
r1p	R_{1p}	JWL coefficient for component 1 products
r2p	R_{2p}	JWL coefficient for component 1 products
op	ω_p	JWL coefficient for component 1 explosive
ae	A_e	JWL coefficient for component 1 explosive
be	B_e	JWL coefficient for component 1 explosive
r1e	R_{1e}	JWL coefficient for component 1 explosive
r2e	R_{2e}	JWL coefficient for component 2 explosive
oe	ω_e	JWL coefficient for component 1 explosive
t0	T_0	Ambient temperature (˚K)
mu	μ	Viscosity
lambda	λ	Thermal conductivity (cal/cm microsec deg)
cp	c_p	specific heat
r0	r_0	pore size (radius -- cm.)
phi	ϕ	
eac	E_{ac}	Activation energy
k0a	K_{0a}	Arrhenius prefactor
qa	Q_a	Heat of reaction
eqa	E_{qa}	Energy of detonation
rhoa	ρ_a	
igt	*igt*	Reaction growth type.
ha	H_a	
xa	X_a	
ya	Y_a	
za	Z_a	
fm1	f_{m1}	Mass fraction for explosive component. f_{m1}: EQ. 1.0: one component explosive, input completed LT. 1.0: two component explosive

This equation of state models the shock initiation and reaction growth of solid high explosives. The initiation is based on viscoplastic heating in the reactant surrounding microscopic pores. The yield strength is compared with the applied stress. Pore collapse begins if the stress exceeds the strength. The rate of collapse is determined by the pressure and the viscosity. The resulting heating rate is calculated. Sources for this data are handbooks, and porosimetry measurements. If the heating rate is greater than the heat conduction rate the local temperature in the neighborhood of the pores will rise until either the pressure is released, or the condition for runaway reaction is reached.

The heat of reaction and energy of detonation are similar quantities but not identical because they are measured in different ways. The heat of reaction is tabulated in reports on criticality experiments such as the ODTX test. The heat of detonation is obtained from cylinder tests and is usually listed with equation of state data for the product gases.

At criticality, if $IGT = 1$, the reaction proceeds according to the formula

$$\frac{dF}{dt} = H(1-F)^y p^z \tag{5-36}$$

for the global chemical reaction rate. The appropriate values can be obtained from reactive flow Lagrange analysis, or by trial and error fit to whatever appropriate experimental data is available. The formula used for $IGT = 1$ is more or less traditional but has no reliable theoretical basis. Other growth formulas will become available as a user option as our understanding improves.

This model may be used for two component composite explosives. If the mass fraction f_{m1} is less than 1.0, DYNA2D will expect additional input describing the second component to model a two-component composite explosive.

The two reacting explosives decompose with independent reaction rates, but will remain in pressure equilibrium within each element. At present *it is assumed that the most sensitive component is described by first.* Once the first component starts reacting it will ignite the second component.

6.14 Equation-of-State Form 13: Pressure vs. Time via Load Curve

Command	Variable	Description
lc	*load_curve*	Load curve defining pressure as a function of time

This is a special purpose equation-of-state wherein the pressure in a material is specified *a priori* as a function of time. Thus, this is not a true equation-of-state in the traditional sense since the pressure in the material does not depend on its deformation history. This model has been found useful for simplified models of reacting fluids and in other special modeling circumstances.

6.15 Equation-of-State Form 14: JWLB

Command	Variable	Description
a1 ... a5	$A_1 \ldots A_5$	
r1 ... r5	$R_1 \ldots R_5$	
al1 ... al5	$A_{\lambda 1} \ldots A_{\lambda 5}$	
bl1 ... bl5	$B_{\lambda 1} \ldots B_{\lambda 5}$	
rl1 ... rl5	$R_{\lambda 1} \ldots R_{\lambda 5}$	
c	C	
omega	ω	
e0	E_8	Initial internal energy per unit initial volume

This equation-of-state, Jones-Wilkens-Lee-Baker (JWLB), is

$$p = \sum_{i=1}^{5} A_i \left[1 - \frac{\lambda}{R_i V}\right] e^{-R_i V} + \frac{\lambda E}{V} + C\left(1 - \frac{\lambda}{\omega}\right) V^{-(\omega+1)} \tag{5-37}$$

where

$$\lambda = \sum_{i=1}^{5} (A_{\lambda i} V + B_{\lambda i}) e^{-R_{\lambda i} V} + \omega . \tag{5-38}$$

In the above, V is relative volume, E is energy per unit initial volume, and A_i, R_i, $A_{\lambda i}$, $B_{\lambda i}$, $R_{\lambda i}$, C, and ω are material constants.

7 NIKE2D MATERIAL PROPERTY COMMANDS

This portion of the manual contains MAZE commands used to specify NIKE2D material model parameters. Included with these descriptions are brief theoretical explanations of the particular model. To obtain the references mentioned in this manual section, the user is referred to the "NIKE2D A Nonlinear, Implicit, Two-Dimensional Finite Element Code for Solid Mechanics - User Manual".

7.1 General Material Definition Commands

The following commands will override the default falues for the current material only.

head *heading*	Material identification. Command is entered in the form: head *heading*.
ro ρ	Material density
thick *thickness*	Thickness - plane stress analysis
mrtemp *temperature*	Material reference temperature
stiff α	Stiffness coefficient - Rayleigh damping

7.2 NIKE2D Material Type 1: Elasticity

Command	Variable	Definition
e	*E*	Young's modulus
v	*v*	Poisson's ratio

This model produces isotropic, linear elastic material behavior.

7.3 NIKE2D Material Type 2: Orthotropic Elasticity

Command	Variable	Description
e	E	Young's modulus
v	v	Poisson's ratio
ea	E_a	Elastic modulus, E_a
eb	E_b	Elastic modulus, E_b
ec	E_c	Elastic modulus, E_c
prba	v_{ba}	Poisson's ratio, v_{ba}
prca	v_{ca}	Poisson's ratio, v_{ca}
prcb	v_{cb}	Poisson's ratio, v_{cb}
gab	G_{ab}	Shear modulus
aopt	*option*	Material axes *option*: 0.0: locally orthotropic with axes determined by angle Ψ and element nodes n_1 and n_2 specified on each element card 1.0: locally orthotropic with axes determined by a point in space and the global location of Gauss integration points 2.0: globally orthotropic with axes determined by angle Ψ_G.
yp	r_p	Coordinate r_p (defined for AOPT = 1.0)
zp	z_p	Coordinate z_p (defined for AOPT = 1.0)
psig	Ψ_G	Angle Ψ_G (radians; defined for AOPT = 2.0)

See DYNA2D Material Type 2 for theoretical description, page 112.

7.4 NIKE2D Material Type 3: Kinematic/Isotropic Elastic-Plastic

Command	Variable	Description
e	E	Young's modulus
v	v	Poisson's ratio
sigy	σ_o	Yield stress
etan	E_p	Hardening modulus
beta	β	Hardening parameter
npts	n	Number of points in stress-effective plastic strain curve
eps	$\varepsilon_1 \ldots \varepsilon_n$	Effective plastic strain
es	$\sigma_1 \ldots \sigma_n$	Effective stress

The material behavior is elastoplastic and includes either linear or nonlinear strain hardening. The specification of hardening parameter β, where $0 \le \beta \le 1$, results in either kinematic, isotropic, or a combination of kinematic and isotropic linear hardening, as shown in Figure 4.2. Purely kinematic or purely isotropic hardening is obtained by setting β to 0.0 or 1.0, respectively. The linear hardening law has the form

$$\sigma_y = \sigma_o + \beta E_p \bar{\varepsilon}^p \tag{6-1}$$

where σ_y is the current yield stress and the effective plastic strain $\bar{\varepsilon}^p$ is given by

$$\bar{\varepsilon}^p = \int^t d\bar{\varepsilon}^p. \tag{6-2}$$

Increment $d\bar{\varepsilon}^p$ defined in terms of the plastic strain tensor $d\varepsilon_{ij}^p$ as

$$d\bar{\varepsilon}^p = \left(\frac{2}{3} d\varepsilon_{ij}^p d\varepsilon_{ij}^p\right)^{\frac{1}{2}}. \tag{6-3}$$

For isotropic hardening, the effective stress σ is given by

$$\sigma = \left(\frac{3}{2} s_{ij} s_{ij}\right)^{\frac{1}{2}}, \tag{6-4}$$

where s_{ij} is the deviatoric stress tensor. For kinematic hardening,

$$\sigma = \left(\frac{3}{2} \eta_{ij} \eta_{ij}\right)^{\frac{1}{2}}, \tag{6-5}$$

where the translated stress η_{ij} is defined as

$$\eta_{ij} = s_{ij} - \alpha_{ij}, \tag{6-6}$$

and α_{ij} is the back stress tensor. The hardening modulus E_p, which represents the slope of the yield stress σ_y vs. effective plastic strain $\bar{\varepsilon}^p$ curve, can be written in terms of the tangent modulus E_T as

$$E_p = \frac{E E_T}{E - E_T}. \tag{6-7}$$

The tangent modulus E_T is the slope of the inelastic portion of a uniaxial stress vs. strain curve (or equivalently the effective stress σ vs. effective strain ε curve). An isotropic, nonlinear hardening law may be specified by defining the $\sigma_y - \bar{\varepsilon}^p$ relationship. For this case, $\bar{\varepsilon}_1^p$ must be zero.

Kinematic and isotropic hardening models yield identical behavior under monotonic loading. Under cyclic loading, kinematic hardening predicts a hysteretic energy dissipation, while isotropic hardening predicts no energy dissipation after the first cycle. The isotropic model is somewhat faster in computation time, however.

7.5 NIKE2D Material Type 4: Thermo-Elastic-Plastic

Command	Variable	Description
npts	n	Number of temperature values for which material constants are defined ($n \leq 8$).
temp	$T_1 \ldots T_n$	Temperatures
e	$E_1 \ldots E_n$	Young's modulus at T_i
v	$\nu_1 \ldots \nu_n$	Poisson's ratio at T_i
alpha	$\alpha_1 \ldots \alpha_n$	Secant coefficients of thermal expansion
sigy	σ_o	Yield stress at T_i
etan	$E_{t1} \ldots E_{tn}$	Hardening modulus at T_i

The material behavior is elastoplastic with isotropic, linear strain hardening. Material parameters E, ν, σ_0, and E_p can all be functions of temperature. Thermal expansion due to temperature change is included when nonzero values of $\bar{\alpha}$ are specified. The secant coefficient of thermal expansion $\bar{\alpha}$ can also be a function of temperature. Total thermal strain ε_{ij}^T is defined in terms of the secant coefficient $\bar{\alpha}$ as

$$\varepsilon_{ij}^T = \bar{\alpha}(T - T_{ref})\delta_{ij} \ , \tag{6-8}$$

where T is the current temperature and T_{ref} is the material reference temperature. Therefore, temperature dependent, secant coefficients of thermal expansion $\bar{\alpha}$ should be defined as the value **to** that temperature, not the value **at** that temperature. The secant coefficient $\bar{\alpha}$ is related to the tangent coefficient of thermal expansion α at a temperature T by

$$\bar{\alpha}(T) = \frac{1}{T - T_{ref}} \int_{T_{ref}}^{T} \alpha(T) dT \ . \tag{6-9}$$

For temperature **independent** coefficients of thermal expansion, $\bar{\alpha}$ is identical to α, and the classical definition of thermal expansion is valid.

If all material parameters are temperature independent, one set of parameters may be defined with T_1 set to zero. This case commonly occurs when thermal expansion is the only temperature related effect of interest. In all other cases, at least two temperatures and material parameters must be specified. The analysis will be terminated if a material temperature falls outside the defined range.

7.6 NIKE2D Material Type 5: Soil and Crushable Foam

Command	Variable	Description
g	G	Shear modulus
ku	K_u	Bulk unloading modulus
a0	a_0	Yield function constant, a_0
a1	a_1	Yield function constant, a_1
a2	a_2	Yield function constant, a_2
pc	p_{cut}	Pressure cutoff
uopt	*option*	Unloading *option*: 0.0: volumetric crushing 1.0: no volumetric crushing
n	$N_1 \ldots N_n$	Hardening exponent at T_i
npts	n	Number of points in volumetric strain vs. pressure curve
vs	$e^v{}_1 \ldots e^v{}_n$	Volumetric strain
p	$p_1 \ldots p_n$	Pressures

This model, derived from the HONDO code (see Key [1974]), allows for the specification of a nonlinear pressure p vs. volumetric strain ε^v relationship (Figure 4.3). Pressure is positive in compression and volumetric strain is negative in compression. Tabulated data should be specified in order of increasing compression. Volumetric crushing is optional, and represents the unloading behavior illustrated in Figure 4.3. For volumetric crushing, the unloading modulus is K_u. Without volumetric crushing, unloading follows the specified pressure-volumetric strain curve. A minimum pressure p_c $(p_c < 0)$ may also be defined. If the pressure drops below this cutoff value, it is reset to the cutoff value.

Deviatoric behavior is elastic, perfectly plastic, with a pressure dependent yield function ϕ

$$\phi = J_2 - (a_0 + a_1 p + a_2 p^2), \tag{6-10}$$

where

$$J_2 = \frac{1}{2} s_{ij} s_{ij}. \tag{6-11}$$

For nonzero a_1 and a_2, the flow is nonassociative. On the yield surface, $J_2 = (1/3)\sigma_y^2$, where σ_y is the uniaxial yield stress (or equivalently the effective yield stress). Therefore, constants a_0, a_1, and a_2 may be determined from uniaxial test data and

$$\sigma_y = [3(a_0 + a_1 p + a_2 p^2)]^{\frac{1}{2}} . \tag{6-12}$$

7.7 NIKE2D Material Type 6: Viscoelasticity

Command	Variable	Description
k	K	Bulk modulus
gg	G_G	Instantaneous shear modulus
gr	G_R	Long term shear modulus
tp	β	Time parameter: mflag .EQ. 0: decay constant b mflag .EQ. 1: time relaxation constant t
mflag	*flag*	Model formulation *flag*: EQ. 0: standard NIKE2D formulation EQ. 1: Kelvin viscoelastic formulation

Two types of viscoelastic formulations are available. For the standard NIKE2D formulation (MFLAG = 0.0), the deviatoric stress rate $\dot{s}_{ij}$ is given by

$$\dot{s}_{ij} = 2\int_0^t G(t-\tau)\dot{e}_{ij}d\tau , \tag{6-13}$$

where the shear relaxation modulus $G(t)$ is defined by

$$G(t) = G_R + (G_G - G_R)e^{-\beta t} , \tag{6-14}$$

and $\dot{e}_{ij}$ is the deviatoric strain rate. The evolution of components of deviatoric stress for the Kelvin formulation (MFLAG = 1.0) is governed by

$$\dot{s}_{ij} + \frac{1}{\tau}s_{ij} = (1 + \delta_{ij})G_G\dot{e}_{ij} + (1 + \delta_{ij})\frac{G_R}{\tau}e_{ij} \qquad \text{(no sum)}. \tag{6-15}$$

These computed components are projected into the deviatoric stress space.

The Kelvin formulation is primarily suited for shear response such as seismic applications. For each of the viscoelastic formulations, the volumetric response is elastic, $p = -K\varepsilon^v$.

7.8 NIKE2D Material Type 7: Thermal-Orthotropic Elasticity

Command	Variable	Description
ea	e_a	Elastic modulus, e_a
eb	e_b	Elastic modulus, e_b
ec	e_c	Elastic modulus, e_c
vba	v_{ba}	Poisson's ratio, v_{ba}
vca	v_{ca}	Poisson's ratio, v_{ca}
vcb	v_{cb}	Poisson's ratio, v_{cb}
alpa	α_a	Coefficient of thermal expansion, α_a
alpb	α_b	Coefficient of thermal expansion, α_b
alpc	α_c	Coefficient of thermal expansion, α_c
gab	G_{ab}	Shear modulus
aopt	*option*	Material axes option: EQ. 0.0: locally orthotropic with material axes determined by angle Ψ and element nodes n_1 and n_2 specified on each element card EQ. 1.0: locally orthotropic with material axes determined by a point in space and the global location of Gauss integration points EQ. 2.0: globally orthotropic with material axes determined by angle Ψ_G.
yp	r_p	Coordinate r_p (defined for AOPT = 1.0)
zp	z_p	Coordinate z_p (defined for AOPT = 1.0)
psig	Ψ_G	Angle Ψ_G. (radians; defined for AOPT = 2.0)

This model is similar to Material Type 2, except that orthotropic thermal expansion is included with the specification of α_a, α_a, and α_a.

7.9 NIKE2D Material Type 8: Thermoelastic Creep

Command	Variable	Description
npts	n	Number of temperature values for which material constants are defined
temp	$T_1 \ldots T_n$	Temperatures
g	$G_1 \ldots G_n$	Shear modulus at T_i
k	$K_1 \ldots K_n$	Bulk modulus at T_i
alpha	$\alpha_1 \ldots \alpha_n$	Secant coefficients of thermal expansion
a	$a_1 \ldots a_n$	Creep coefficient at T_i
b	$\beta_1 \ldots \beta_n$	Creep exponent at T_i

The implementation of this model was developed by Krieg (1977), and includes both thermoelastic and creep behavior. The instantaneous creep rate $\dot{\varepsilon}$ is given by a power law of the form

$$|\dot{\varepsilon}| = a|\sigma|^b , \tag{6-16}$$

where σ is effective stress. Material parameters *G*, *K*, *a*, and *b* can all be functions of temperature. Thermal expansion due to temperature change is included when nonzero values of $\bar{\alpha}$ are specified. The secant coefficient of thermal expansion $\bar{\alpha}$ can also be a function of temperature. Total thermal strain ε_{ij}^T is defined in terms of the secant coefficient $\bar{\alpha}$ as

$$\varepsilon_{ij}^T = \bar{\alpha}(T - T_{ref})\delta_{ij} , \tag{6-17}$$

where T is the current temperature and T_{ref} is the material reference temperature (see Material Type 4 for additional information). At least two temperatures and their corresponding material parameters must be specified. The analysis will be terminated if a material temperature falls outside the defined range.

7.10 NIKE2D Material Type 9: Blatz-Ko Rubber

Command	Variable	Description
g	G	Shear modulus

See DYNA2D Material Model Type 7 for theoretical description, page 123.

7.11 NIKE2D Material Type 10: Power Law Elastic-Plastic with Failure

Command	Variable	Description
e	E	Young's modulus
v	v	Poisson's ratio
k	K	Strength coefficient
n	N	Hardening exponent
sigt	σ_{tf}	Strength to failure from tensile test
sigs	σ_{sf}	Strength to failure from shear test
shear	*lshear*	Load curve for shear strength to failure as a function of pressure
ifail	*flag*	Failure flag option: EQ. 0.0: failure not modeled EQ. 1.0: Mohr-Coulomb EQ. 2.0: Drucker-Prager EQ. 3.0: Mohr-Coulomb and Drucker-Prager

The material behavior is elastoplastic with isotropic, power law hardening. The nonlinear hardening law has the form

$$\sigma_y = k(\varepsilon_0 + \bar{\varepsilon}^p)^n \,, \tag{6-18}$$

where ε_0 is the initial yield strain given by

$$\varepsilon_0 = \left(\frac{E}{k}\right)^{\frac{1}{n-1}} \,. \tag{6-19}$$

Failure is included if IFAIL is specified as nonzero. When failure is reached at all Gauss points in an element, the strength of the element is reduced to nearly zero, and that element is deleted in the binary plot database. Several failure options are available and are defined as follows:

- If IFAIL = 1.0 and load curve number LSHEAR = 0.0, the Mohr-Coulomb failure criterion is invoked. Failure is initiated when stress values exceed the Mohr-Coulomb criterion

$$\sigma_{sf} = m\sigma_{tf} + b \,. \tag{6-20}$$

The slope *m* and itercept *b* are determined from tensile and shear test values, σ_{tf} and σ_{sf}, by

$$m = \frac{\frac{1}{2}\sigma_{tf} - \sigma_{sf}}{\sqrt{\sigma_{tf}\sigma_{sf} - \sigma_{sf}^2}}, \tag{6-21}$$

$$b = \frac{\sigma_{tf}\sigma_{sf}}{2\sqrt{\sigma_{tf}\sigma_{sf} - \sigma_{sf}^2}} = m\frac{\sigma_{tf}\sigma_{sf}}{\sigma_{tf} - 2\sigma_{sf}}. \tag{6-22}$$

The values of σ_{tf} and σ_{sf} are restricted to

$$\sigma_{sf} > 0, \tag{6-23}$$

$$\sigma_{tf} > \sigma_{sf}. \tag{6-24}$$

- If IFAIL = 1.0 and load curve number LSHEAR is nonzero, a modified Mohr-Coulomb criterion is used. Failure is initiated when the maximum shear stress exceeds the appropriate value from the shear strength to failure vs. pressure curve.
- If IFAIL = 2.0, the Drucker-Prager failure criterion is invoked. The Drucker-Prager surface is constructed from the tensile and shear strengths at failure, σ_{tf} and σ_{sf}.
- If IFAIL = 3.0, both the Drucker-Prager and the appropriate Mohr-Coulomb failure criteria are invoked.

7.12 NIKE2D Material Type 11: Creep Plasticity

Command	Variable	Description
e	E	Young's modulus
v	ν	Poisson's ratio
t	T_0	Initial temperature (˚K)
rcv	ρC_v	Density specific heat
b	β	Hardening parameter
c1	C_1	Rate dependent yield stress coefficient
c2	C_2	Rate dependent yield stress exponent
c3	C_3	Rate independent yield stress coefficient
c4	C_4	Rate independent yield stress exponent
c5	C_5	Transition coefficient
c6	C_6	Transition exponent
c7	C_7	Hardening coefficient
c8	C_8	Hardening exponent
c9	C_9	Dynamic recovery coefficient
c10	C_{10}	Dynamic recovery exponent
c11	C_{11}	Diffusion recovery coefficient
c12	C_{12}	Diffusion recovery exponent

See DYNA2D Material Model Type 14 for theoretical description, page 137.

7.13 NIKE2D Material Type 12: Power Law Thermo-Elastic-Plastic

Command	Variable	Description
npts	n	Number of temperature values for defined material constants
temp	$T_1 \ldots T_n$	Temperatures
e	$E_1 \ldots E_n$	Young's modulus at T_i
v	$\nu_1 \ldots \nu_n$	Poisson's ratio at T_i
alpha	$\alpha_1 \ldots \alpha_n$	Secant coefficients of thermal expansion
k	$K_1 \ldots K_n$	Strength coefficient at T_i
n	$N_1 \ldots N_n$	Hardening exponent at T_i

The material behavior is elastoplastic with isotropic, power law hardening. The nonlinear hardening law has the form

$$\sigma_y = k(\varepsilon_0 + \bar{\varepsilon}^p)^n \ , \tag{6-25}$$

where ε_0 is the initial yield strain given by

$$\varepsilon_0 = \left(\frac{E}{k}\right)^{\frac{1}{n-1}} \ . \tag{6-26}$$

Material parameters E, ν, k, and n can all be functions of temperature. Thermal expansion due to temperature change is included when nonzero values of α are specified. The coefficient of thermal expansion α can also be a function of temperature and, for this model, represents a tangent value. Thermal strain rate $\dot{\varepsilon}_{ij}^T$ is defined in terms of the tangent coefficient of thermal expansion α as

$$\dot{\varepsilon}_{ij}^T = \alpha(T)\dot{T}\delta_{ij} \ , \tag{6-27}$$

where T is the current temperature.

If all material parameters are temperature independent, one set of parameters may be defined with T_1 set to zero. This case commonly occurs when thermal expansion is the only temperature related effect of interest. In all other cases, at least two temperatures and their corresponding material parameters must be specified. The analysis will be terminated if a material temperature falls outside the defined range.

7.14 NIKE2D Material Type 13: Strain Rate Dependent Isotropic Elastic-Plastic

Command	Variable	Description
e	$E_1 \ldots E_n$	Young's modulus at T_i
v	$v_1 \ldots v_n$	Poisson's ratio at T_i
curve	*load_curve*	Load curve number for yield stress vs. strain rate
ep	E_p	Hardening modulus

The material behavior is elastoplastic with strain rate dependent, isotropic hardening. The hardening law has the form

$$\sigma_y = \sigma_0(\dot{\varepsilon}) + E_p \bar{\varepsilon}^p \ , \tag{6-28}$$

where $\bar{\varepsilon}^p$ is the effective plastic strain and σ_0 is determined by the load curve specification of σ_0 vs. $\dot{\varepsilon}$. The effective strain rate $\dot{\varepsilon}$ is defined as

$$\dot{\varepsilon} = \left(\frac{2}{3}\dot{e}_{ij}\dot{e}_{ij}\right)^{\frac{1}{2}} \ , \tag{6-29}$$

where $\dot{e}_{ij}$ is the deviatoric strain rate tensor. The hardening modulus E_p, which represents the slope of the yield stress σ_y vs. effective plastic strain $\bar{\varepsilon}^p$ curve, can be written in terms of the tangent modulus E_T as

$$E_p = \frac{E E_T}{E - E_T}. \tag{6-30}$$

The tangent modulus E_T is the slope of the inelastic portion of a uniaxial stress vs. strain curve.

7.15 NIKE2D Material Type 14: Circumferentially Cracked Elastoplasticity

Command	Variable	Description
e	E	Young's modulus$_i$
v	ν	Poisson's ratio
sigy	σ_o	Yield stress
etan	E_p	Hardening modulus$_i$
beta	β	Elastic modulus: $0 \le \beta \le 1$ coefficient - Rayleigh damping
npts	n	Number of points in stress-effective plastic strain curve
eps	$\varepsilon_1 \ldots \varepsilon_n$	Effective plastic strain
es	$\sigma_1 \ldots \sigma_n$	Effective stress

This model is applicable only to axisymmetric structures. The material behavior is identical to Material Type 3, except that elements of this material cannot carry tensile circumferential stress. By specifying an initial relative volume greater than one on the element cards, the development of compressive circumferential stresses can be delayed or prevented since a gap must close circumferentially before compressive hoop stress develops.

7.16 NIKE2D Material Type 15: Extended Two Invariant Geologic Cap Model

Command	Variable	Description
k	K	Initial bulk modulus
g	G	Initial shear modulus
alpha	α	Failure envelope parameter
theta	Θ	Failure envelope linear coefficient
gamma	γ	Failure envelope exponential coefficient
b	β	Failure envelope exponent
r	R	Cap surface axis ratio
d	D	Hardening law exponent
w	W	Hardening law coefficient
x0	X_0	Hardening law parameter

Command	Variable	Description
cbar	$\bar{c}$	Kinematic hardening coefficient
n	N	Kinematic hardening parameter
iplot	*flag*	Plot database flag: EQ. 1.0: Hardening variable EQ. 2.0: Cap - J_1 axis intercept $X_{(\kappa)}$ EQ. 3.0: Volumetric plastic strain ε^p_v EQ. 4.0: First stress invariant J_1 EQ. 5.0: Second stress invariant $\sqrt{J_{2D}}$ EQ. 8.0: Response mode number EQ. 9.0: Number of iterations
lckv	*load_curve*	Load curve number for K / K_0 vs. V / V_0
lcgv	*load_curve*	Load curve number for G / G_0 vs. V / V_0
lcrv	*load_curve*	Load curve number for R / R_0 vs. V / V_0
lctv	*load_curve*	Load curve number for T / T_0 vs. V / V_0
itype	*flag*	Formulation flag: EQ. 1.0: Soil or concrete (cap surface may contact) EQ. 2.0: Rock (cap surface may not contact)
ivect	*flag*	Vectorization flag: EQ. 0.0: Vectorized (fixed number of iterations) EQ. 1.0: Fully iterative
t	T	Tension cutoff, $T< 0$ (positive in compression)

NIKE2D allows the material parameters K, G, R, and T to be functions of relative volume V/V_0. Load curves are used to describe the variation of each parameter with relative volume, and these curves must pass through the point (1.0,1.0). If a load curve number is specified as 0.0, that material parameter is held constant at its initial value throughout the analysis. See DYNA2D Material Type 18 for theoretical description, page 147.

7.17 NIKE2D Material Type 16: Ramberg-Osgood Elastic-Plastic

Command	Variable	Description
gy	γ_y	Reference shear strain
ty	τ_y	Reference shear stress
alpha	α	Stress coefficient
r	r	Stress component
k	k	Bulk modulus

The Ramberg-Osgood equation is an empirical constitutive relation to represent the one-dimensional elastic-plastic behavior of many materials. This implementation of the model was developed by Whirley and Engelmann (1991b), and allows a simple rate independent representation of the hysteretic energy dissipation observed in materials subjected to cyclic shear deformation. For monotonic loading, the stress-strain relationship is given by

$$\frac{\gamma}{\gamma_y} = \frac{\tau}{\tau_y} + \alpha\left|\frac{\tau}{\tau_y}\right|^r \quad \text{if } \gamma > 0, \tag{6-31}$$

$$\frac{\gamma}{\gamma_y} = \frac{\tau}{\tau_y} - \alpha\left|\frac{\tau}{\tau_y}\right|^r \quad \text{if } \gamma < 0, \tag{6-32}$$

where γ is the shear strain and τ is the shear stress. The model approaches perfect plasticity as the stress exponent $r \to \infty$. These equations must be augmented to correctly model unloading and reloading material behavior. The first load reversal is detected by $\gamma\dot{\gamma} < 0$. After the first reversal, the stress-strain relationship is modified to

$$\frac{\gamma - \gamma_0}{2\gamma_y} = \frac{\tau - \tau_0}{2\tau_y} + \alpha\left|\frac{\tau - \tau_0}{2\tau_y}\right|^r \quad \text{if } \gamma > 0, \tag{6-33}$$

$$\frac{\gamma - \gamma_0}{2\gamma_y} = \frac{\tau - \tau_0}{2\tau_y} - \alpha\left|\frac{\tau_0 - \tau}{2\tau_y}\right|^r \quad \text{if } \gamma < 0, \tag{6-34}$$

where γ_0 and τ_0 represent the values of strain and stress at the point of load reversal. Subsequent load reversals are detected by $(\gamma - \gamma_0)\dot{\gamma} < 0$.

The Ramberg-Osgood equations are inherently one-dimensional, and are assumed to apply to shear components. To generalize this theory to the multidimensional case, it is assumed that each component of the deviatoric stress and deviatoric tensorial strain is independently related by the one-dimensional stress-strain equations. The computed stress is projected onto the deviatoric stress space. The volumetric behavior is elastic, and therefore the pressure p is found by

$$p = -K\varepsilon^{v}, \tag{6-35}$$

where ε^{v} is the volumetric strain.

7.18 NIKE2D Material Type 17: Thermo-Elastic-Plastic with 8 Curves

Command	Variable	Description
npts	n	Number of temperature values for defined material constants
temp	$T_1 \ldots T_n$	Temperatures
e	$E_1 \ldots E_n$	Young's modulus at T_i
v	$\nu_1 \ldots \nu_n$	Poisson's ratio at T_i
alpha	$\alpha_1 \ldots \alpha_n$	Secant coefficients of thermal expansion
lc	$lc_1 \ldots lc_n$	Load curve number for yield stress σ_i vs. effective plastic strain ε^{p}_{i} at T_i

This model is similar to Material Type 4, except that a nonlinear strain hardening law is used in place of linear hardening. Material parameters E and ν, as well as the nonlinear hardening law, are functions of temperature. Load curves are used to specify the nonlinear hardening law by defining yield stress σ_y vs. effective plastic strain $\bar{\varepsilon}_p$, for each temperature. The effective hardening law used for the stress update procedure is found by a suitable interpolation. Note that the first point on each $\sigma_y - \bar{\varepsilon}^p$ load curve must be $\bar{\varepsilon}^p = 0$.

Thermal expansion due to temperature change is included when nonzero values of $\bar{\alpha}$ are specified. The secant coefficient of thermal expansion $\bar{\alpha}$ can also be a function of temperature. Total thermal strain ε^{T}_{ij} is defined in terms of the secant coefficient $\bar{\alpha}$ as

$$\varepsilon^{T}_{ij} = \bar{\alpha}(T - T_{ref})\delta_{ij}\ , \tag{6-36}$$

where T is the current temperature and T_{ref} is the material reference temperature (see Material Type 4 for additional information).

At least two temperatures, and their corresponding material parameters and load curve numbers, must be specified. The analysis will be terminated if a material temperature falls outside the defined range.

7.19 NIKE2D Material Type 18: Thermo-Elastic-Plastic Quench

Command	Variable	Description
k	*k*	Bulk modulus
lflag	*flag*	Load curve option flag: EQ. 0.0: identical points for each curve EQ. 1.0: different points for each curve
lpq	*lphs1*	Load curve number - phase 1
lp2	*lphs2*	Load curve number - phase 2
e1	E_1	Phase 1 load curve number for Young's modulus
v1	v_1	Phase 1 load curve number for Poisson's ratio
a1	α_1	Phase 1 load curve number for secant coefficients of thermal expansion
sig1	σ_1	Phase 1 load curve number for yield stress
ep1	E_{p1}	Phase 1 load curve number for hardening modulus
e2	E_2	Phase 2 load curve number for Young's modulus
v2	v_2	Phase 2 load curve number for Poisson's ratio
a2	α_2	Phase 2 load curve number for secant coefficients of thermal expansion
sig2	σ_2	Phase 2 load curve number for yield stress
ep2	E_{p2}	Phase 2 load curve number for hardening modulus
e3	E_3	Phase 3 load curve number for Young's modulus
v3	v_3	Phase 3 load curve number for Poisson's ratio
a3	α_3	Phase 3 load curve number for secant coefficients of thermal expansion
sig3	σ_3	Phase 3 load curve number for yield stress
ep3	E_{p3}	Phase 3 load curve number for hardening modulus

This model is similar to Material Type 4, except that it allows properties to be defined for up to three different phases. The number of phases and the fractional amounts of each phase are determined from load curves LPHS1 and LPHS2, and the current temperature T as follows:

- For a three phase material, load curve LPHS1 defines the fraction of phase one vs. temperature and load curve LPHS2 defines the fraction of phase two vs. temperature. The fraction of phase three, at any temperature, is 1.0 minus the sum of the fractions of phase one and phase two, at that temperature.
- For a two phase material, load curve LPHS1 defines the fraction of phase 1 vs. temperature and LPHS2 should be set to zero. The fraction of phase two is 1.0 minus the fraction of phase one.
- For a one phase material, LPHS1 and LPHS2 should be set to zero. The fraction of phase one is 1.0.

Material parameter load curves specify the material parameters E, ν, σ_0, and E_p as a function of temperature, for each phase. For the stress update and thermal expansion computation, material properties are first determined for each phase at the current temperature. Properties are then averaged according to the fractional amounts of each phase. The bulk modulus K is only a characteristic modulus and is not used for the stress update calculation.

Thermal expansion due to temperature change is included when nonzero values of $\bar{\alpha}$ are specified for any active phase. The "phase weighted" average of the temperature dependent secant coefficients of thermal expansion $\bar{\alpha}$ may be denoted $\bar{\alpha}_A$. Total thermal strain ε_{ij}^T is then defined in terms of the averaged secant coefficient $\bar{\alpha}_A$ as

$$\varepsilon_{ij}^T = \bar{\alpha}_A(T - T_{ref})\delta_{ij} , \tag{6-37}$$

where T is the current temperature and T_{ref} is the material reference temperature (see Material Type 4 for additional information).

Material parameter load curve numbers must be specified for each material parameter of each of the phases considered. Each load curve must contain at least two temperatures and their corresponding material parameter values. The analysis will be terminated if a material temperature falls outside the defined range.

7.20 NIKE2D Material Type 19: Strain Rate Sensitive Power Law Elastic-Plastic

e	*E*	Young's modulus
v	*v*	Poisson's ratio
k	*k*	Strength coefficient
n	*n*	Hardening exponent
m	*m*	Strain rate sensitivity exponent
isr	*rate*	Initial strain rate. Default: 0.0002

This model differs from material type 10 in that the hardening law is strain rate sensitive, and the hardening parameters may be a function of effective plastic strain. The material behavior is elasto-plastic with isotropic hardening. The nonlinear hardening law is given by

$$\sigma_y = k\dot{\varepsilon}^m(\varepsilon_0 + \bar{\varepsilon}^p) \ , \tag{6-38}$$

where $\dot{\varepsilon}$ is the effective strain rate and ε_0 is the initial yield strain given by

$$\varepsilon_0 = \left(\frac{E}{k\dot{\varepsilon}^m}\right)^{\frac{1}{n-1}} \ . \tag{6-39}$$

Absence of strain hardening can be modeled by setting the hardening exponent to a very small positive value, i.e. 0.0001.

7.21 NIKE2D Material Type 20: Power Law Thermo-Elastic-Plastic with Failure

npts	n	Number of temperature values for defined material constants
temp	$T_1 \ldots T_n$	Temperatures
e	$E_1 \ldots E_n$	Young's modulus at T_i
v	$\nu_1 \ldots \nu_n$	Poisson's ratio at T_i
alpha	$\alpha_1 \ldots \alpha_n$	Secant coefficients of thermal expansion
k	$K_1 \ldots K_n$	Strength coefficient at T_i
n	$N_1 \ldots N_n$	Hardening exponent at T_i
sigt	$\sigma_{tf1} \ldots \sigma_{tfn}$	Strength to failure from tensile test
sigs	$\sigma_{sf1} \ldots \sigma_{sfn}$	Strength to failure from shear test
ifail	*flag*	failure flag option: EQ. 1.0: Mohr-Coulomb EQ. 2.0: Drucker-Prager EQ. 3.0: Mohr-Coulomb and Drucker-Prager

This model is identical to Material Type 12, except that simulation of material failure is included. Prior to failure, the material behavior is elastoplastic with isotropic power law hardening. Material parameters E, ν, k, and n can all be functions of temperature. Thermal expansion due to temperature change is included when nonzero values of α are specified. The coefficient of thermal expansion α can also be a function of temperature and, for this model, represents a tangent value. Thermal strain rate $\dot{\varepsilon}_{ij}^{T}$ is defined in terms of the tangent coefficient of thermal expansion α as

$$\dot{\varepsilon}_{ij}^{T} = \alpha(T)\dot{T}\delta_{ij} , \tag{6-40}$$

where T is the current temperature.

Failure is initiated when material state variables exceed the specified criterion. When failure is reached at all Gauss points in an element, the strength of the element is reduced to nearly zero, and that element is deleted from the plot database. The specified tensile and shear strengths at failure may be functions of temperature. In this case, effective values for σ_{tf} and σ_{sf} are found for the current temperature by interpolation. Several failure options are available and are defined as follows:

- If IFAIL = 1.0, the Mohr-Coulomb failure criterion is invoked. Failure is initiated when stress values exceed the Mohr-Coulomb criterion

$$\sigma_{sf} = m\sigma_{tf} + b\,. \tag{6-41}$$

The slope m and intercept b are determined from interpolated tensile and shear test values σ_{tf} and σ_{sf} by

$$m = \frac{\frac{1}{2}\sigma_{tf} - \sigma_{sf}}{\sqrt{\sigma_{tf}\sigma_{sf} - \sigma_{sf}^2}}\,, \tag{6-42}$$

$$b = \frac{\sigma_{tf}\sigma_{sf}}{2\sqrt{\sigma_{tf}\sigma_{sf} - \sigma_{sf}^2}} = m\frac{\sigma_{tf}\sigma_{sf}}{\sigma_{tf} - 2\sigma_{sf}}\,. \tag{6-43}$$

The values of σ_{tf} and σ_{sf} are restricted to

$$\sigma_{sf} > 0\,, \tag{6-44}$$

$$\sigma_{tf} > \sigma_{sf}\,. \tag{6-45}$$

and therefore specified values at each temperature are also restricted.

- If IFAIL = 2.0, the Drucker-Prager failure criterion is invoked. The Drucker-Prager surface is constructed from tensile and shear strengths at failure σ_{tf} and σ_{sf}.
- If IFAIL = 3.0, both the Mohr-Coulomb and Drucker-Prager failure criteria are invoked.

If all material parameters (including failure parameters), are temperature independent, one set of parameters may be defined with T_1 set to zero. This case commonly occurs when thermal expansion is the only temperature effect of interest. In all other cases, at least two temperatures and their corresponding material and failure parameters must be specified. The analysis will be terminated if a material temperature falls outside the defined range.

7.22 NIKE2D Material Type 21: Nonlinear Elastic-Plastic

k0	K_0	Constant bulk modulus
k1	K_1	Bulk modulus coefficient
g0	G_0	Constant shear modulus
g1	G_1	Shear modulus coefficient
y0	Y_0	Constant yield stress
y1	Y_1	Yield stress coefficient
g	G	Initial shear modulus
pc	p_c	Minimum pressure; $p_c < 0$

This model incorporates a pressure-volume relationship of the form

$$p = \frac{K_0}{K_1}\left[\left(\frac{V_0}{V}\right)^{K_1} - 1\right] \tag{6-46}$$

for hydrostatic compression ($p > 0$), and

$$p = K_0 ln\left(\frac{V_0}{V}\right) \tag{6-47}$$

for hydrostatic tension ($p \geq 0$). Equivalently, the volumetric response may be written as

$$p = -K\varepsilon^v, \tag{6-48}$$

where the effective bulk modulus K is

$$K = K_0 + K_1 p \tag{6-49}$$

for hydrostatic compression and

$$K = K_0, \tag{6-50}$$

for hydrostatic tension. A minimum pressure p_c, ($p_c < 0$) may be specified. If the drops below this cutoff value, it is reset to the cutoff value.

The deviatoric response is elastic, perfectly plastic. The effective shear modulus G and yield stress σ_y are defined as

$$G = G_0 + G_1 p, \tag{6-51}$$

$$\sigma_y = Y_0 + Y_1 p, \tag{6-52}$$

for hydrostatic compression and

$$G = G_0, \tag{6-53}$$

$$\sigma_y = Y_0, \tag{6-54}$$

for hydrostatic tension.

7.23 NIKE2D Material Type 22: Polynomial Hyperelastic Rubber

Command	Variable	Description
c100 .. c400	$C_{100} ... C_{400}$	Strain Energy Density Coefficients
c010 c020	$C_{010} C_{020}$	Strain Energy Density Coefficients
c110 c210	$C_{110} C_{210}$	Strain Energy Density Coefficients
c001 c101	$C_{001} C_{101}$	Strain Energy Density Coefficients

The implementation of this model was developed by Kenchington (1988), and is defined in a total Lagrangian context. The strain energy density function W is a polynomial form of Green-Lagrange strain E_{ij}. It is defined as

$$W = C_{100}I_1 + C_{200}I_1^2 + C_{300}I_1^3 + C_{400}I_1^4 + C_{010}I_2 + C_{020}I_2^2 + C_{110}I_1I_2 + C_{210}I_1^2I_2 + C_{001}I_3 + C_{101}I_1I_3 \tag{6-55}$$

where I_1, I_2, and I_3 are the first, second, and third principal invariants of the Green-Lagrange strain tensor E_{ij}. The second Piola-Kirchhoff stress S_{ij} is found by differentiating the strain energy density W with respect to E_{ij}, (6-56)

$$S_{ij} = \frac{\partial W}{\partial E_{ij}}. \tag{6-57}$$

The Cauchy stress τ_{ij} is determined from the second Piola-Kirchhoff stress S_{ij} by

$$\tau_{ij} = \frac{1}{V_r} F_{ik} F_{jl} S_{kl}, \tag{6-58}$$

where V_r is the relative volume and F_{ij} is the deformation gradient.

7.24 NIKE2D Material Type 23: Primary, Secondary, Tertiary Creep

Command	Variable	Description
e	E	Young's modulus
v	v	Poisson's ratio
a	A	Stress coefficient
n	n	Stress exponent
m	m	Time exponent

The implementation of this model was developed by Whirley and Henshall (1990). The effective creep strain ε^c is defined as

$$\varepsilon^c = A\sigma^n \bar{t}^m, \tag{6-59}$$

where $\bar{t}$ is the effective time, and the effective stress σ is defined as

$$\sigma = \sqrt{\frac{3}{2}\sigma_{ij}\sigma_{ij}}. \tag{6-60}$$

The effective creep strain evolves in the direction of current deviatoric stress and the volumetric behavior is assumed elastic. By varying the value of the time exponent m, the three classical creep regimes may be simulated:

- $m < 1$ Primary creep
- $m = 1$ Secondary (steady-state) creep
- $m > 1$ Tertiary creep

7.25 NIKE2D Material Type 24: Deformation Mechanism

npts	*n*	Number of temperature values for defined material constants
temp	$T_1 \ldots T_n$	Temperatures
e	$E_1 \ldots E_n$	Young's modulus at T_i
v	$\nu_1 \ldots \nu_n$	Poisson's ratio at T_i
alpha	$\alpha_1 \ldots \alpha_n$	Secant coefficients of thermal expansion
dmid	*id*	Deformation mechanism *id*: EQ. 1: Obstacle controlled plasticity EQ. 2: Power law creep EQ. 3: Power law breakdown EQ. 4: Diffusional flow
iplot	*flag*	Plot database *flag*: EQ. 0: Effective plastic strain EQ. 1: Thermal strain EQ. 2: Effective plastic strain rate EQ. 3: Strength EQ. 4: Grain size
tol	*tolerance*	Constitutive convergence tolerance
iter	*iterations*	Maximum number of constitutive iterations
nket	*n*	Number of kinetic equation temperatures
nspt	*n*	Number of strength parameter temperatures
ngst	*n*	Number of grain size temperatures

NOTE: Material Type #24 requires additional material model data. Refer to the "NIKE2D A Nonlinear, Implicit, Two-Dimensional Finite Element Code for Solid Mechanics - User Manual" for additional information.

7.26 NIKE2D Material Type 25: Gurson-Tvergaard Void Growth Plasticity

Command	Variable	Description
e	E	Young's modulus
v	v	Poisson's ratio
sig0	σ_o	Yield stress
ep	E_p	Hardening modulus
q1	q_1	First Gurson parameter
q2	q_2	Second Gurson parameter
f0	f_0	Initial void fraction

The material behavior is elastoplastic with modifications to include void growth under hydrostatic tension. The development and implementation of this model is discussed in Whirley and Engelmann (1991), and is based on a unification of the Gurson and Tvergaard formulations described in Hom and McMeeking (1989). Studies have shown that ductile fracture in metals may be proceeded by the generation of considerable porosity, and thus the presence of voids lead to yield behavior dependent on the hydrostatic component of stress. The observed behavior is usually attributed to void growth, and the matrix material is believed essentially incompressible. The Gurson and Tvergaard theories macroscopically model the void growth phenomenon, and may be useful for problems with regions of large hydrostatic tensile stresses, such as at the base of notches and in the constrained tensile loading of bars and plates.

The Gurson and Tvergaard yield surfaces are members of the family defined by the function

$$\phi = \frac{\sigma^2}{\sigma_y^2} + 2 f q_1 \cosh\left(\frac{\sigma_{kk}}{2\sigma_y}\right) - (1 + q_2 f^2) \ , \tag{6-61}$$

where $q_1 = q_2 = 1.0$ for the Gurson model, and $q_1 = 1.5$ and $q_2 = q_1^2$ for the Tvergaard model. The von Mises yield surface of material type 3 may be recovered by setting $q_1 = q_2 = 0$. The effective stress σ is defined as the norm of the deviatoric stress tensor s_{ij},

$$\sigma = \left(\frac{3}{2} s_{ij} s_{ij}\right)^{\frac{1}{2}} \ . \tag{6-62}$$

Linear isotropic hardening is assumed so the current yield stress σ_y is given by

$$\sigma_y = \sigma_0 + E_p \bar{\varepsilon}^p , \tag{6-63}$$

where $\bar{\varepsilon}^p$ is the effective plastic strain. Associated flow is considered, and therefore the rate of plastic strain is in the direction normal to the yield surface,

$$\dot{\varepsilon}_{ij}^p = \dot{\lambda} \frac{\partial \phi}{\partial \sigma_{ij}} . \tag{6-64}$$

The evolution of the void fraction f is given by

$$\dot{f} = (1 - f)\varepsilon_{kk}. \tag{6-65}$$

Typical initial void fractions are in the range of $0.001 \le f_0 \le 0.1$.

8 TOPAZ2D - CHEMICAL TOPAZ2D MATERIAL PROPERTY COMMANDS

This portion of the manual contains MAZE commands used to specify TOPAZ2D and CHEMICAL TOPAZ2D material model parameters. Included with these descriptions are brief theoretical explanations of the particular model. To obtain the references mentioned in this manual section, the user is referred to the “TOPAZ2D A Nonlinear, Explicit, Two-Dimensional Finite Element Code for Solid Mechanics - User Manual”.

8.1 General Material Definition Commands

The following commands will override the default falues for the current material only.

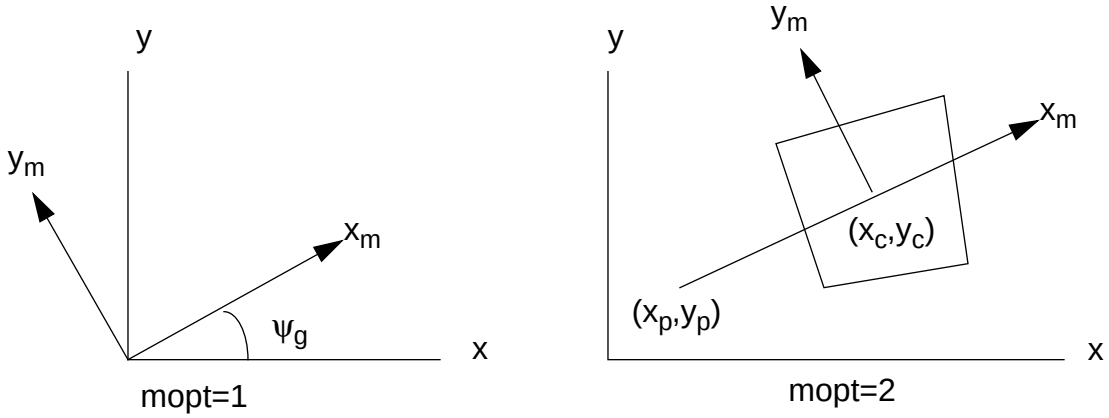
mat *n* *heading*	Material identification consisting of material number *n* and a *heading*. Command is entered in the form: mat *n* *heading*.
mt *type*	Material type
den *density*	Material density
tlat $T_{transition}$	Transition temperature
hlat ΔH	Latent heat of transition
mgen *curve* mgen *multiplier*	Thermal generation rate *curve* number and thermal generation rate *multiplier*
mang ψ_g	Material angle (degrees) for an orthotropic material
tref T_{ref}	Internal energy reference temperature - CHEMICAL TOPAZ2D
q0 Q_0	Reference internal energy - CHEMICAL TOPAZ2D
done *value*	Doneness value - CHEMICAL TOPAZ2D
gasv *value*	Gas value - CHEMICAL TOPAZ2D
frev volume	Free volume - CHEMICAL TOPAZ2D
mixc $n\ m_1 \ldots m_n$	Mixture material composition - CHEMICAL TOPAZ2D
mixs $n_1\ n_2\ mn_1 \ldots mn_2$	Mixture material composition - CHEMICAL TOPAZ2D

8.2 Material Type 1: Isotropic

Command	Variable	Description
cp	*1* c_{p1}	Heat capacity
con1	*1* k_1	Thermal conductivity

8.3 Material Type 2: Orthotropic

Command	Variable	Description
cp	*1* c_{p1}	Heat capacity
con1	*1* k_1	Thermal conductivity - x direction
con2	*1* k_2	Thermal conductivity - y direction
mopt	*1* Ψ_g	Orthotropic material angle (degrees) from (x,y) global coordinate counter-clockwise from respective coordinates
	or	or
	2 x_p y_p	Center point of local coordinate system



8.4 Material Type 3: Isotropic, Temperature Dependent

Command	Variable	Description
temp	$n\ T_1 \ldots T_n$	Temperature; n ≤ 8
cp	$n\ c_{p1} \ldots c_{pn}$	Heat capacity; n ≤ 8
con1	$n\ k_{11} \ldots k_{1n}$	Thermal conductivity; n ≤ 8

8.5 Material Type 4: Orthotropic, Temperature Dependent

Command	Variable	Description
con2	$n\ k_{21} \ldots k_{2n}$	Single value conductivity in y-direction; n ≤ 8

This material type combines the command sets of Material Type 2 and Type 3 with the addition of command “con2” denoting conductivity in the y-direction.

8.6 Material Type 5: Isotropic, Temperature Dependent from Material Data Base

Command	Variable	Description
nc	n_c	Number of heat capacity data points NOTE: If $n_c > 1$, command TCP5 is required
nk	n_k	Number of thermal conductivity data points NOTE: If $n_k > 1$, command TCN5 is required
tc0	T_{c0}	Heat capacity
cp0	c_{p0}	Thermal conductivity
tcp5	$n_c\ T_1\ c_{p1} \ldots$ $T_n\ c_{pn}$	n: number of specific heat pairs; 2 ≤ n ≤ 24 T_1: initial temperature c_{p1}: initial specific heat T_n : final temperature c_{pn}: final specific heat
tcn5	$n_k\ T_1\ k_1 \ldots$ $T_n\ k_n$	n: number of conductivity pairs; 2 ≤ n ≤ 24 T_1: initial temperature c_{p1}: initial conductivity T_n : final temperature c_{pn}: final conductivity

8.7 Material Type 6: Power Law

Command	Variable	Description
hcc	$c_0\, c_1\, c_2\, c_3$	Heat capacity coefficients
tcc	$k_0\, k_1\, k_2\, k_3$	Thermal conductivity coefficients

Heat capacity $c = c_0 + c_1T + c_2T^2 + c_3T^3$ and thermal conductivity are $k= k_0 + k_1T + k_2T^2 + k_3T^3$ are represented by a cubit polynomial. The use of this material model is desirable for nonlinear problems because the polynomials are evaluated directly without the use of table lookups.

8.8 Material Type 7: High Temperature Cutoff

8.9 Material Type 8: Low Temperature Cutoff

Command	Variable	Description
temp	$n\ T_1 \ldots T_n$	Temperature; n ≤ 8
cp	$n\ c_{p1} \ldots c_{pn}$	Heat capacity; n ≤ 8
con1	$n\ k_{11} \ldots k_{1n}$	Thermal conductivity; n ≤ 8
tcut	T_{cut}	Temperature cutoff
vcut	v_{cut}	Thermal capacity before cutoff

This material is isotropic, temperature dependent with a temperature cutoff above or below which the material assumes a constant value of thermal conductivity. This allows simulation of materials that are transparent or opaque to heat flow above a high cutoff temperature and below a low cutoff temperature. Suggested values for the thermal conductivity are: a.) $k = k_0 \times 10^2$ W/m^2 K for a thermally transparent material and b.) $k = k_0 \times 10^{-2}$ W/m^2 K for a thermally opaque material. k_0 is a nominal value for the thermal conductivity of the material.

8.10 Material Type 9: Isotropic, Temperature Dependent, Phase Change

Command	Variable	Description
temp	$n\ T_1 \ldots T_n$	Temperature; n ≤ 8
hlat	λ	Latent heat
cp	$n\ c_{p1} \ldots c_{pn}$	Heat capacity; n ≤ 8
con1	$n\ k_{11} \ldots k_{1n}$	Thermal conductivity; n ≤ 8
tsol	$T_{solidus}$	Solidus temperature
tliq	$T_{liquidus}$	Liquidus temperature

TOPAZ will enhance the heat capacity versus temperature values to account for the latent heat such that in the phase change region from T_L to T_S the heat capacity is

$$c(T) = m\left[1 - \cos 2\pi\left(\frac{T - T_S}{T_L - T_S}\right)\right] \qquad T_S < T < T_L$$

where

T_L = liquidus temperature

T_S = solidus temperature

T = temperature

m = multiplier (calculate by TOPAZ) such that $\lambda = \int_{T_S}^{T_L} c(T)dT$

λ = latent heat

c = heat capcity

8.11 Material Type 3000: Temperature Dependent Trump Material Data Base

Command	Description
tmid *id unit*	Select material identification number *id* from Trump database using *units*: EQ.1: SI EQ. 2: CGS
tgrc *curve*	Thermal generation rate curve number
tgrm *factor*	Thermal generation rate multiplication *factor*

9 EXAMPLES OF MAZE INPUT FILES

The following input command files illustrate the principles and structure of MAZE discussed in this manual. The examples incorporate a variety of techniques and MAZE command structures in the construction and design of the resulting mesh together with its accompanying material model and other supportive data. They are presented as guides for the MAZE user and are not intended to be comprehensive in scope.

9.1 DYNA2D: Light Weight Gun Barrel Design

```
1
ppoff lpoff
c nofl
maztl 2.0e-4
c
c dynamic analysis of test
c
c vertical lines
ld 1 lp 2 0.0   -15.0 0.0   75.0   c centerline
ld 2 lp 2 1.875 -15.0 1.875 75.0   c bore in composite
ld 3 lp 2 3.520 -15.0 3.520 75.0   c inside of steel liner
ld 4 lp 1 0.0 -4.5
     lar 4.5 0.0  4.5
     lp 4 3.6 19.5 3.6 20.5 3.6325 20.85 3.6325 75.0  c outside steel liner
ld 5 lp 1 1.875 -6.883
     lar 7.1932 0.0 12.730         c bottom outer composite
ld 6 lp 1 7.6625 0.0
     lvc 105. 5.230878
     lp 2   5.6   19.5   5.6  75.0 c outside of the composite
ld 7 lp 3 9.25 -10.0 9.25  1.784 8.75   3.65 c outside of steel
c horizontal lines
ld 100 lp 2 0.0 -6.883 25.0 -6.883  c backface of composite
ld 101 lp 2 0.0 -6.1   25.0 -6.1    c backface of steel
ld 102 lp 2 0.0  0.0   25.0  0.0    c origin line
ld 103 lp 2 0.0  1.784 25.0  1.784  c begining of steel taper
ld 104 lp 2 0.0  3.65  25.0  3.65   c front of outer steel
ld 105 lp 2 0.0 19.5   15.0 19.5    c begining of weld
ld 106 lp 2 0.0 19.6   15.0 19.6    c end of weld
ld 107 lp 2 0.0 25.0   15.0 25.0    c break in barrel/end of pres.
ld 108 lp 2 0.0 65.5   25.0 65.5    c end of barrel
c the radial transition line
```

```
ld 201 lp 1 0.0  0.0  lvc -45 15.0
c make the steel liner p 1-7
    part 102   1   4   2  1  5   6 y
    part 102   2   4   3  1 12   6 y
    part 102   3   4   4  1  2   4 y
    part 104   3 102   4  1  6  15 y
t12 part 105   3 104   4  1  3 -90 0.2 y
    part 107   3 106   4  1  3  80 y
    part 108   4 107   3  1  3 360 y
c the weld p 8
part 106   3 105   4  2  3  3 y
c the composite shell  p 9 - 14
    part   2   5 201   4  3  13  10 y
    part 201   5 102   4  4  13  14 y
    part 102   6 104   4  5 -16  13 1.5 y
t12 part 105   4 104   6  5   8  -80 0.5 y
    part 105   6 107   4  5   8  -32 0.4 y
    part 107   6 108   4  5   8  -140 0.5 y
c the outer steel p 15 - 17
part 101   7 102   5  6 -6  9 2.2 y
part 102   7 103   6  6  5  5 y
part 103   7 104   6  6  5  6 y
assm
title
Light Weight Gun Barrel Design
c the liner
mg 2 3 p 2 b
s s v mg 2 4
m 1 2 mg 2 5 m 6 7
c
c the weld region
m 2 8 m 6 8
c
c the composite
m 9 10 gm 10 11
p 10 b s
p 11 b ess 3 s mg 11 12 mg 11 13 mg 11 14
c
c the steel counter mass
mg 16 17 gm 15 16
c
c slideline time
c composite to the outer steel
sln 1 3 p 15 b msrs 3 msrs 4 msrs 5 msrs 6 slne off
        p  9 b slvs 1
        p 10 b slvs 1
```

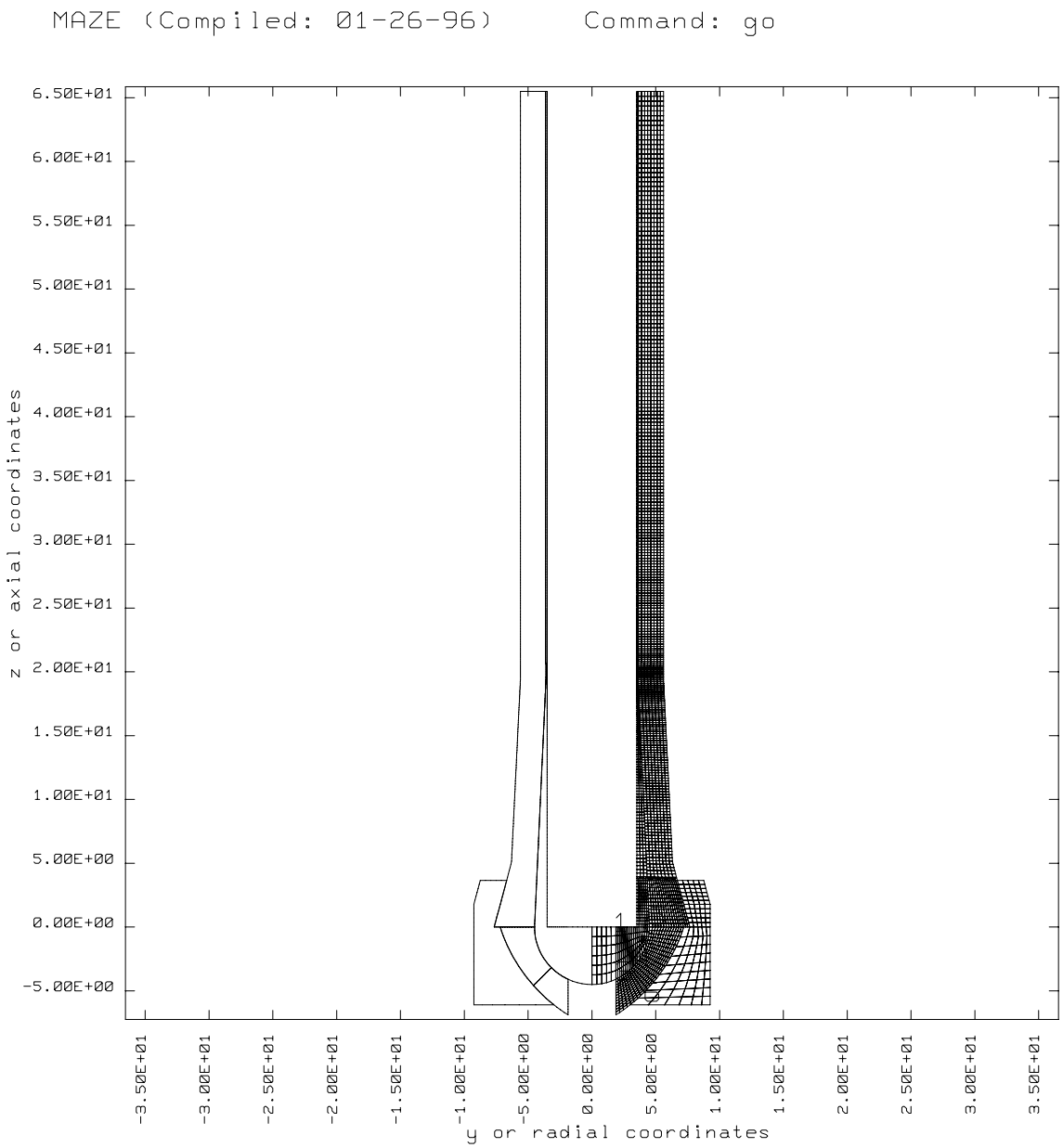
```
        p 11 b slv 2757 3627
c the liner to the composite
sln 2 3 slne off
        p  8 b slvs 2
        p  6 b slvs 2
        p  7 b slvs 2
        p 11 b msrs 4
sln 3 3 slne off
        p  2 b slvs 1
        p 11 b msrs 4
        p 10 b msrs 3
        p  9 b msrs 3
c
c apply the pressure
c
c define the need lookup table
lut 1 67
  -100.0      0.0
  5.5118E-02  0.0000E+00  5.5118E-02  1.0000E-04  1.0630E-01  2.0000E-04
  1.0630E-01  3.0000E-04  1.6142E-01  4.0000E-04  2.1654E-01  5.0000E-04
  2.1654E-01  6.0000E-04  2.6772E-01  7.0000E-04  3.2283E-01  8.0000E-04
  4.2913E-01  9.0000E-04  4.8425E-01  1.0000E-03  5.9055E-01  1.1000E-03
  6.9685E-01  1.2000E-03  8.0709E-01  1.3000E-03  9.6457E-01  1.4000E-03
  1.0748E+00  1.5000E-03  1.2874E+00  1.6000E-03  1.4488E+00  1.7000E-03
  1.7165E+00  1.8000E-03  1.9331E+00  1.9000E-03  2.2559E+00  2.0000E-03
  2.5236E+00  2.1000E-03  2.8976E+00  2.2000E-03  3.2756E+00  2.3000E-03
  3.7047E+00  2.4000E-03  4.1890E+00  2.5000E-03  4.6693E+00  2.6000E-03
  5.2598E+00  2.7000E-03  5.8504E+00  2.8000E-03  6.5512E+00  2.9000E-03
  7.2480E+00  3.0000E-03  8.0512E+00  3.1000E-03  8.8583E+00  3.2000E-03
  9.7717E+00  3.3000E-03  1.0736E+01  3.4000E-03  1.1756E+01  3.5000E-03
  1.2886E+01  3.6000E-03  1.4012E+01  3.7000E-03  1.5248E+01  3.8000E-03
  1.6535E+01  3.9000E-03  1.7933E+01  4.0000E-03  1.9327E+01  4.1000E-03
  2.0831E+01  4.2000E-03  2.2386E+01  4.3000E-03  2.3996E+01  4.4000E-03
  2.5610E+01  4.5000E-03  2.7327E+01  4.6000E-03  2.9098E+01  4.7000E-03
  3.0925E+01  4.8000E-03  3.2803E+01  4.9000E-03  3.4681E+01  5.0000E-03
  3.6614E+01  5.1000E-03  3.8602E+01  5.2000E-03  4.0642E+01  5.3000E-03
  4.2681E+01  5.4000E-03  4.4776E+01  5.5000E-03  4.6870E+01  5.6000E-03
  4.9016E+01  5.7000E-03  5.1161E+01  5.8000E-03  5.3366E+01  5.9000E-03
  5.5618E+01  6.0000E-03  5.7874E+01  6.1000E-03  6.0130E+01  6.2000E-03
  6.2437E+01  6.3000E-03  6.2815E+01  6.3170E-03  200.0       8.0e-3
gun 2 1 18.31
p 1 b  pbcs 3 1 1.0 1.0  nbcs 4 1
p 2 b  pbcs 3 1 1.0 1.0
pbcs 4 1 1.0 1.0
p 6 b  pbcs 4 1 1.0 1.0
p 7 b  pbcs 4 1 1.0 1.0
```

```
p 8 b  pbcs 4 1 1.0 1.0

c load curve : time vs. pressure
lcd 1 65
   0.0000000E+00      0.0
   9.9999910E-05   4162.990
   1.9999994E-04   4829.608
   2.9999996E-04   5568.650
   3.9999999E-04   6383.745
   5.0000002E-04   7278.520
   5.9999991E-04   8255.878
   6.9999998E-04   9318.288
   7.9999998E-04   10473.73
   8.9999993E-04   11766.33
   1.0000000E-03   13186.79
   1.1000000E-03   14742.09
   1.2000001E-03   16438.17
   1.3000000E-03   18280.26
   1.4000000E-03   20271.41
   1.5000000E-03   22413.06
   1.6000000E-03   24704.06
   1.7000001E-03   27140.35
   1.8000001E-03   29714.22
   1.8999999E-03   32414.80
   2.0000001E-03   35226.71
   2.0999999E-03   38130.19
   2.2000002E-03   41101.74
   2.3000001E-03   44113.64
   2.3999999E-03   47134.54
   2.5000002E-03   50130.48
   2.6000000E-03   53065.60
   2.7000001E-03   55903.34
   2.8000001E-03   58607.40
   2.9000000E-03   61143.69
   2.9999998E-03   63481.28
   3.1000003E-03   65593.47
   3.2000002E-03   67458.78
   3.3000002E-03   69061.40
   3.4000000E-03   70392.02
   3.4999999E-03   71447.31
   3.6000004E-03   72229.76
   3.7000002E-03   72582.15
   3.8000001E-03   72041.51
   3.9000001E-03   70976.92
   3.9999997E-03   69544.41
   4.1000005E-03   67846.30
```

```
   4.1999999E-03   65960.96
   4.3000006E-03   63888.54
   4.4000000E-03   61212.05
   4.5000003E-03   58336.87
   4.6000006E-03   55407.11
   4.6999999E-03   52503.63
   4.8000002E-03   49676.78
   4.9000001E-03   46958.20
   5.0000004E-03   44381.42
   5.1000006E-03   41968.51
   5.2000000E-03   39713.64
   5.3000003E-03   37610.60
   5.3999997E-03   35664.88
   5.5000004E-03   33866.77
   5.6000007E-03   32202.47
   5.7000001E-03   30659.65
   5.8000004E-03   29227.58
   5.8999998E-03   27896.08
   6.0000001E-03   26656.31
   6.1000008E-03   25500.44
   6.2000002E-03   24421.04
   6.3000005E-03   23411.76
   6.3170008E-03   23243.83

title
Barrel Analysis
term 6.0e-03
plti 1.4e-05
prti 9.0
rfmts 0.002
teo 0
wbcd dyna2d

c head 4130 steel   rc 32  (142 sig-ys)
mat 1 13
ro 7.32993E-04
e 30.0e+06
pr 0.3
k 2.190E+05
n 8.184E-02
endmat

c head 4130 steel   rc 45 (212 sig-ys)
mat 2 13
ro 7.32993E-04
e 30.0e+06
```

```
pr 0.3
k 3.142E+05
n 8.168E-02
endmat

c head homogenized +-10/+-89/+-89/+-89 t1000  quasi-at end
mat 3 2
ro  1.681e-4
ea  1.2084e+06
eb  7.0011e+06
ec 19.8376e+06
prba  0.327
prca  0.375
prcb  0.078
gab  0.4844e+06
aopt 1
rp 0.0
zp 0.0
psig 0.0
endmat

c head homogenized +-10/+-89/+-89/+-89 t1000 in the bend
mat 4 2
ro  1.681e-4
ea  1.2084e+06
eb  7.0011e+06
ec 19.8376e+06
prba  0.327
prca  0.375
prcb  0.078
gab  0.4844e+06
aopt 1
rp 0.0
zp 0.0
endmat

c head homogenized +-10/+-89/+-89/+-89 t1000  up the barrel
mat 5 2
ro  1.681e-4
ea  1.2084e+06
eb  7.0011e+06
ec 19.8376e+06
prba  0.327
prca  0.375
prcb  0.078
gab  0.4844e+06
```

```
aopt 2
psig 0.0
endmat

c head 4130 steel   rc 30
mat 6 13
ro 7.32993E-04
e 30.0e+06
pr 0.3
k 1.573E+05
n 8.184E-02
endmat
c
end
```


MAZE (Compiled: 01-26-96) Command: go
6.50E+01
6.00E+01
5.50E+01
5.00E+01
4.50E+01
4.00E+01
3.50E+01
3.00E+01
2.50E+01
2.00E+01
1.50E+01
1.00E+01
5.00E+00
0.00E+00
-5.00E+00
z or axial coordinates
-3.50E+01
-3.00E+01
-2.50E+01
-2.00E+01
-1.50E+01
-1.00E+01
-5.00E+00
0.00E+00
5.00E+00
1.00E+01
1.50E+01
2.00E+01
2.50E+01
3.00E+01
3.50E+01
y or radial coordinates

9.2 NIKE2D: Thermomechanical Notched Casting Analysis

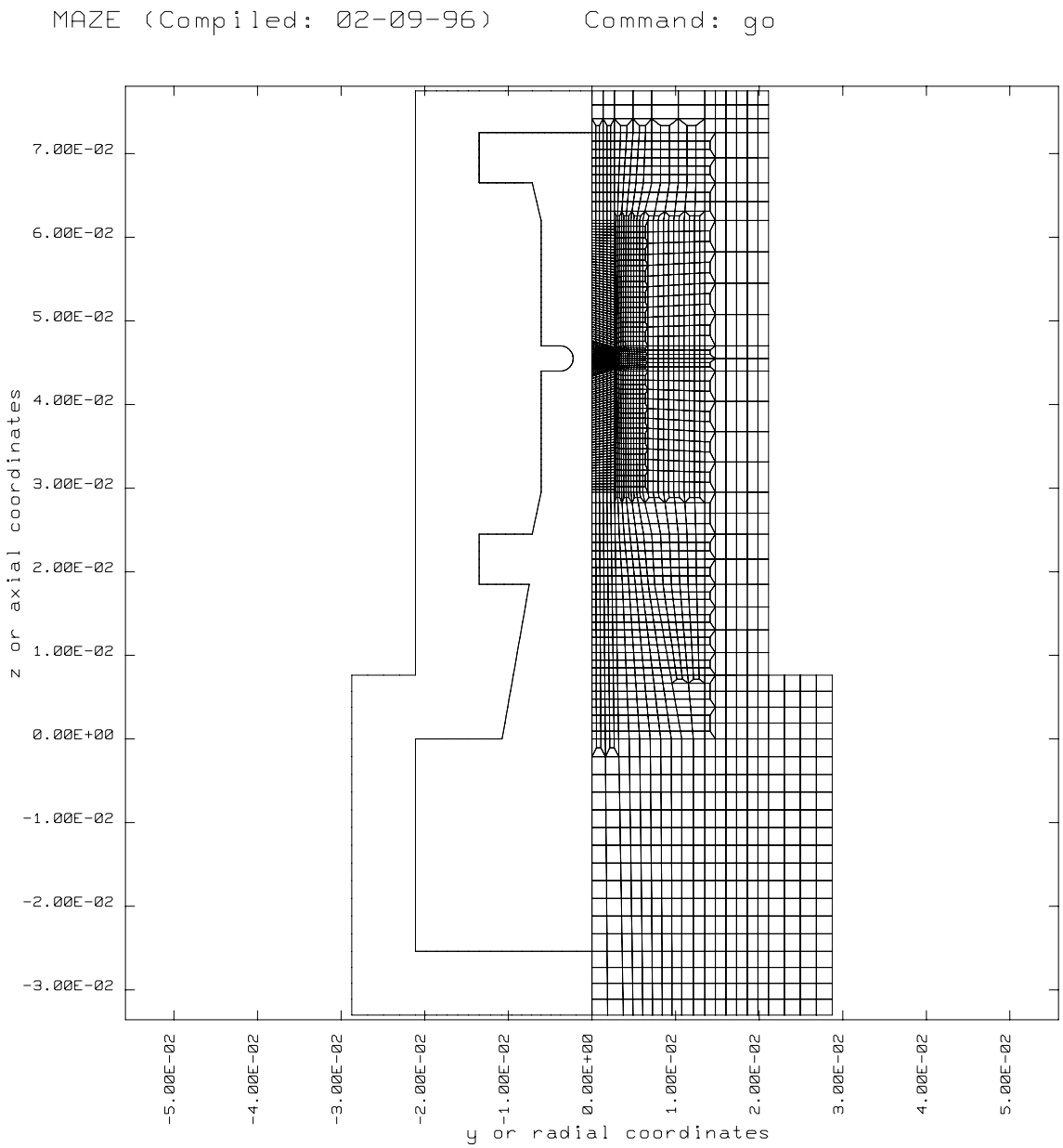
```
1
maztol 1.0e-4
param ra 0.00225;
param rb 0.0015;
param r0 [%ra+%rb];
param r1 [%ra+%rb*(1.-sin(45))];
param z1 [0.0455-%rb];
param z2 [0.0455+%rb];
param z3 [0.0455-%rb*sin(45)];
param r4 [%ra+0.5*%rb];
param z4 [0.0455-0.5*%rb];
param z5 [0.0455+0.5*%rb];
param z6 [0.0455+%rb*sin(45)];
param z7 [0.0455-%r0*sin(30)];
param z8 [0.0455+%r0*sin(30)];
c
ld 1 lp 2 0.0 -0.03302 0.0 0.0775
ld 2 lp 3 [%r0]  -0.03302 [%r1]   0.0295  [%r1]   0.0775
ld 3 lp 3 [%r1]   [%z3]   [%r4]   [%z4]   [%r4]   0.0455
ld 4 lp 3 [%r4]   0.0455  [%r4]   [%z5]   [%r1]   [%z6]
ld 5 lp 3 0.0     [%z7]   [%r1]   [%z3]   [%r4]   [%z4]
ld 6 lp 3 0.0     [%z8]   [%r1]   [%z6]   [%r4]   [%z5]
ld 7 lp 2 [%r4]   [%z4]   [%r4]   [%z5]
ld 8 lp 8
0.01075 -0.03302
0.01075  0.0
0.0075   0.0185
0.00715  0.0245
0.006075 0.0295
0.006075 0.0620
0.00715  0.0665
0.00715  0.0775
ld  9 lp 2 0.0135  -0.03302  0.0135   0.0775
ld 10 lp 2 0.02112 -0.03302  0.02112  0.0775
ld 11 lp 2 0.02874 -0.03302  0.02874  0.00762
ld 16 lp 2 0.0     -0.03302  0.02874 -0.03302
ld 17 lp 2 0.0     -0.0254   0.02874 -0.0254
lcc 1 [%r0] [0.0455] 179 270 [%rb] lvc 0.0 [0.02112-%r0]
ld 19 lp 2 0.0 0.0455 0.02112 0.0455
lcc 1 [%r0] [0.0455] 181  90 [%rb] lvc 0.0 [0.02112-%r0]
ld 21 lp 2 0.0  0.00762 0.02874  0.00762
ld 22 lp 2 0.0  0.0     0.02874  0.0
ld 23 lp 2 0.0  0.0185  0.02112  0.0185
```

```
ld 24 lp 2 0.0  0.0245  0.02112  0.0245
ld 25 lp 2 0.0  0.0295  0.02112  0.0295
ld 26 lp 2 0.0  0.0620  0.02112  0.0620
ld 27 lp 2 0.0  0.0665  0.02112  0.0665
ld 28 lp 2 0.0  0.0725  0.02112  0.0725
ld 29 lp 2 0.0  0.0775  0.02112  0.0775
c
part  5 18 19  1  1 12  9 y
part 19 20  6  1  1 12  9 y
t12 part 25  2  5  1  1   6 -48  1.25  y
t12 part 26  1  6  2  1   6 -48  1.25  y
t12 part  8 18  2 25  1 -24 -12  1.25  2.0  y
t12 part  8 26  2 20  1 -24 -12  0.8  2.0  y
part 24  2 25  1  1   6   4  y
t12 part 24  8 25  2  1  -6   4 0.5  y
part 26  2 27  1  1   6   4  y
t12 part 27  2 26  8  1  -6   4 2.0  y
part 23  2 24  1  1   6   6  y
part 23  8 24  2 -1   6   6  1.0 1.0 2.0 1.0 y
part 23  9 24  8  1   6   6  y
part 28  1 27  2  1   6   6  y
part 28  2 27  8 -1   6   6  1.0 1.0 0.5 1.0  y
part 28  8 27  9  1   6   6  y
part 21  2 23  1  1   6  12  y
part 21  8 23  2  1   6  12  y
part 22  2 21  1  1   6   8  y
part 22  8 21  2  1   6   8  y
t13  part 17  2 22  1  1 2  12  y
part 17  8 22  2  1 6  12  y
part 17  9 22  8  1  2  12  y
part 17 10 22  9  1  6  12  y
mg 1  2  mg 1  3  mg 1  4  mg 1  5  mg 1  6
mg 1  7  mg 1  8  mg 1  9  mg 1 10  mg 1 11
mg 1 12  mg 1 13  mg 1 14  mg 1 15  mg 1 16
mg 1 17  mg 1 18  mg 1 19  mg 1 20  mg 1 21
mg 1 22  mg 1 23  mg 1 24
c
t31 part 18  5  7  19  2  9  3  y
t31 part 20 19  7   6  2  9  3  y
part 18  8 19 -3  2 -12 6  0.5  3  y
part 19  8 20 -4  2 -12 6  0.5 3  y
t12 part  9 20  8 18   2  6 12  y
t12 part  9 18  8 25  -2  12 12    1.0  1.0 0.8 1.0  y
t12 part  9 26  8 20  -2  12 12    1.0 1.0 1.25 1.0  y
t12 part 24  9 25  8   2   6  4    y
t12 part 27  8 26  9   2   6  4    y
```

```
t13 part 10 20  9 18   2   2  6    y
t13 part 10 18  9 25   2   4  6    y
t13 part 10 26  9 20   2   4  6    y
t12 part 10 25  9 24   2   2  6    y
t13 part 10 24  9 23   2   2  6    y
part 21  9 23  8   2   6  12   y
t13 part 10 23  9 21   2   4  6    y
t13 part 22  9 21  8   2   2  8   y
t12 part 10 21  9 22   2   4  6    y
part 22 11 21 10  2  4  4   y
t12 part 10 27  9 26   2   2  6    y
t13 part 10 28  9 27   2   2  6    y
part 10 29  9 28   2  3  6   y
t13 part 29  8 28  9   2   2  3   y
t13 part 29  2 28  8   2   2  3   y
t13 part 29  1 28  2   2   2  3   y
part 17 11 22 10  2  4  12  y
part 16  2 17  1  2  2   4  y
part 16  8 17  2  2  6   4  y
part 16  9 17  8  2  2   4  y
part 16 10 17  9  2  6   4  y
part 16 11 17 10  2  4   4  y
mg 25 26  mg 25 27  mg 25 28  mg 25 29  mg 25 30
mg 25 31  mg 25 32  mg 25 33  mg 25 34  mg 25 35
mg 25 36  mg 25 37  mg 25 38  mg 25 39  mg 25 40
mg 25 41  mg 25 42  mg 25 43  mg 25 44  mg 25 45
mg 25 46  mg 25 47  mg 25 48  mg 25 49  mg 25 50
mg 25 55  mg 25 54  mg 25 53  mg 25 52  mg 25 51
assm
c
c ******************************************************************************
rcon  0.0
zcon -0.03302
c
sln 1 3
p 1 b
p 25 b
slbp 1 25
laf 1  atn 0.05
c
lcd 1 2       0.0 2.137E+11     5000. 2.137E+11
lcd 2 2       0.0 3.300E-01     5000. 3.300E-01
lcd 3 2       0.0 1.368E-05     5000. 1.368E-05
lcd 4 14
      273. 8.778E-78
      473. 1.454E-43
```

```
      673. 1.105E-29
      873. 3.666E-22
      973. 1.463E-19
     1073. 1.911E-17
     1173. 1.088E-15
     1273. 3.282E-14
     1373. 6.029E-13
     1473. 7.459E-12
     1573. 6.702E-11
     1673. 4.631E-10
     1773. 2.574E-09
     1873. 1.191E-08
lcd 5 2       0.0  0.188679     5000.  0.188679
lcd 6 2       0.0       0.0     5000.       0.0
lcd 7 2       0.0       1.0     5000.       1.0
c
title
Inc 718 cast notch C
teo -2
iepd 2
af 0
bwmo 1
delt 2.0
dtmax 2.0
dtmin 1.0e-3
term 1000.0
tssf 0.2
maxsteps 10000
maxtries 10
nibsr 10
msrf  25
numref 2
ngoodsteps 2
prti 600
plti 3
dctol  5e-4
dstol  5e-4
wbcd nike2d
c
c ******************************************************************************
mat 1 2000
    1   280.0000E+000.0000E+000.0000E+000.0000E+00
IN718 Specimen
         1         2         3         2         0         0         0         3
 2.095e+11   1.0e+06         4         5         6         7
blank
```

```
blank
blank
endmat
c
mat 2 4
head
Ceramic Mold
npts 5
temp  0.000E+00 2.930E+02 1.600E+03 2.200E+03 5.000E+03
e     5.500E+09 5.500E+09 6.500E+09 1.000E+09 1.000E+06
pr    2.500E-01 2.500E-01 2.500E-01 2.500E-01 2.500E-01
alpha 1.000E-08 1.000E-08 1.000E-08 1.000E-08 1.000E-08
sigy  7.000E+06 7.000E+06 1.680E+07 1.040E+07 5.000E+06
etan  5.500E+03 5.500E+03 5.500E+03 5.500E+03 5.500E+03
endmat
end
```


MAZE (Compiled: 02-09-96) Command: go
z or axial coordinates
7.00E-02
6.00E-02
5.00E-02
4.00E-02
3.00E-02
2.00E-02
1.00E-02
0.00E+00
-1.00E-02
-2.00E-02
-3.00E-02
-5.00E-02
-4.00E-02
-3.00E-02
-2.00E-02
-1.00E-02
0.00E+00
1.00E-02
2.00E-02
3.00E-02
4.00E-02
5.00E-02
y or radial coordinates

9.3 TOPAZ2D: Thermal Analysis

```
1
{
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*  AEAG MARK VII THERMOS                                                            *
*  Maze file: W.MAZ.THERMOS                                                         *
*  Units: Inches                                                                    *
*  Lines generated by Filbert Diaz 2/7/96                                           *
*  Mesh generation by Dr. Robert A. Bailey 2/16/96                                  *
*  Topaz application by Tom Healy 6/3/96                                            *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
This is an example of the transient thermal response of a Thermos filled with a
hot fluid.  The Thermos is initally at 70F and the fluid is at 180F.  Heat is
transferred internal to the Thermos by conduction and radiation.  Heat is
transferred external from the Thermos by natural convection and radiation to the
surrounding environment.  The bottom of the Thermos is adiabatic.
}

lpoff ppoff
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Vacuum-Insulated glass liner                                                    *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld   1 lp 1  0.0000    0.8750
       lap   1.5000    2.3750    0.0000    2.3750      c r=    1.5000 ang= 90.00
       lp 1  1.5000    8.6008
       lap   1.3225    9.2633    0.1750    8.6008      c r=    1.3250 ang= 30.00
       lp 1  0.9590    9.8929
       lap   0.8250   10.3929    1.8250   10.3929      c r=    1.0000 ang= 30.00
       lp 1  0.8250   10.7503
ld   2 lp 1  0.8250   10.7503
       lap   0.5763   10.7627    0.7003   10.7503      c r=    0.1247 ang=174.29
       lp 1  0.5763   10.7627
ld   3 lp 1  0.0000    1.2000
       lap   1.1750    2.3750    0.0000    2.3750      c r=    1.1750 ang= 90.00
       lp 1  1.1750    8.6008
       lap   1.0410    9.1008    0.1750    8.6008      c r=    1.0000 ang= 30.00
       lp 1  0.7046    9.6836
       lap   0.5433   10.4330    1.7871   10.3086      c r=    1.2500 ang= 35.71
       lp 2  0.5763   10.7627    0.5766   10.7661
ld   4 lp 1  0.0000    0.9150
       lap   1.4600    2.3750    0.0000    2.3750      c r=    1.4600 ang= 90.00
       lp 1  1.4600    8.6008
       lap   1.2878    9.2433    0.1750    8.6008      c r=    1.2850 ang= 30.00
       lp 1  0.9243    9.8729
       lap   0.7850   10.3929    1.8250   10.3929      c r=    1.0400 ang= 30.00
       lp 1  0.7850   10.7503
ld   5 lp 1  0.7850   10.7503
       lap   0.6161   10.7588    0.7003   10.7503      c r=    0.0847 ang=174.29
       lp 1  0.6161   10.7588
ld   6 lp 1  0.0000    1.1600
       lap   1.2150    2.3750    0.0000    2.3750      c r=    1.2150 ang= 90.00
       lp 1  1.2150    8.6008
       lap   1.0757    9.1208    0.1750    8.6008      c r=    1.0400 ang= 30.00
       lp 1  0.7392    9.7036
       lap   0.5831   10.4290    1.7871   10.3086      c r=    1.2100 ang= 35.71
```

```
       lp 1  0.6161   10.7588
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Jacket (Plastic)                                                            *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld  11 lp 2  1.6400    1.0000    1.6400    9.2980
ld  12 lp 2  1.5150    9.2980    1.6400    9.2980
ld  13 lp 2  1.5150    9.2980    1.5150   10.1100
ld  14 lp 2  1.4850   10.1100    1.5150   10.1100
ld  15 lp 1  1.4850   10.1100
       lap   0.7927   10.8600    0.4725    9.8699      c r=    1.0406 ang= 58.74
       lp 1  0.7596   10.8600
ld  16 lp 2  1.5000    1.3750    1.5000    8.6008
       lap   1.3225    9.2633    0.1750    8.6008      c r=    1.3250 ang= 30.00
       lp 1  0.9590    9.8929
       lap   0.8250   10.3929    1.8250   10.3929      c r=    1.0000 ang= 30.00
       lp 1  0.8250   10.7503
       lap   0.7596   10.8600    0.7003   10.7503      c r=    0.1247 ang= 61.60
ld  17 lp 2  1.5000    1.3750    1.5770    1.3750
ld  18 lp 2  1.5770    1.0000    1.5770    1.3750
ld  19 lp 2  1.5770    1.0000    1.6400    1.0000
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Tip Protector (Rubber)                                                       *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld  21 lp 2  0.3750    0.7108    0.7500    0.7108
ld  22 lp 2  0.7500    0.7108    0.7500    1.0760
ld  23 lp 1  0.3750    0.9226
       lap   0.7500    1.0760    0.0000    2.3750      c r=    1.5000 ang= 15.52
ld  24 lp 2  0.3750    0.7108    0.3750    0.9226
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Liner Support (Steel)                                                  *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld  31 lp 2  1.0536    0.1250    1.0882    0.1250
ld  32 lp 2  1.0882    0.1250    0.7500    0.7108
ld  33 lp 2  0.7500    0.7108    0.3750    0.7108
ld  34 lp 2  0.3750    0.7108    0.3750    0.8108
ld  35 lp 2  0.3450    0.8108    0.3750    0.8108
ld  36 lp 2  0.3450    0.6808    0.3450    0.8108
ld  37 lp 2  0.7327    0.6808    0.3450    0.6808
ld  38 lp 2  1.0536    0.1250    0.7327    0.6808
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Jacket, Bottom (Plastic)                                                    *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld  41 lp 2  0.0000    0.0000    1.6400    0.0000
ld  42 lp 2  1.6400    0.0000    1.6400    1.0000
ld  43 lp 2  1.5770    1.0000    1.6400    1.0000
ld  44 lp 2  1.5770    1.0000    1.5770    1.3750
ld  45 lp 2  1.5000    1.3750    1.5770    1.3750
ld  46 lp 2  1.0882    0.1750    1.2500    0.1750
       lap   1.5000    0.4250    1.2500    0.4250      c r=    0.2500 ang= 90.00
       lp 1  1.5000    1.3750
ld  47 lp 2  1.0882    0.1250    1.0882    0.1750
ld  48 lp 2  0.0000    0.1250    1.0882    0.1250
c
c    Cork (Portuguese Cork)
c
ld  51 lp 2  0.0000   10.0000    0.5000   10.0000
ld  52 lp 2  0.5000   10.0000    0.6250   11.2500
ld  53 lp 2  0.0000   11.2500    0.6250   11.2500
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

```
c *  Cup (Plastic)                                                          *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld  61 lp 2  1.5150    9.2980    1.6400    9.2980
ld  62 lp 2  1.6400    9.2980    1.6400   10.2980
       lap   1.0000   12.2980   -1.5264   10.3872      c r=    3.1676 ang= 38.72
ld  63 lp 2  0.0000   12.2980    1.0000   12.2980
ld  64 lp 2  0.0000   12.1730    0.9370   12.1730
ld  65 lp 2  1.5150    9.2980    1.5150   10.4746
       lap   0.9370   12.1730   -1.5264   10.3872      c r=    3.0426 ang= 34.29
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Fluid Level (Water)                                                    *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld  71 lvc   0.0000    7.0000    0.000    1.17500
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Construction lines for mesh generation                                 *
c *  Start with line 100 and are added as needed                            *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Fluid                                                                  *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld 100 lp 2   0.0000    0.0000   0.0000   12.2980
ld 101 lvc   0.0000    2.3750    0.0000    1.6400
     part   101  100    3    3    1   15   15  yes                 c part  1
     part   101    3   71  100    1   30   60  yes                 c part  2
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Air over Fluid                                                         *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld 102 lpil    51  100 lpil    51   52 lvc   0.0000    1.1400
ld 103 lvc   0.0000   10.1443    0.0000      0.6000
ld 104 lvc   0.0000   10.4330    0.0000      0.6000
ld 105 lpil    52  103 lpil     3  104
     part    71    3 -102  100    2   30   30   27  yes             c part  3
t13  part   103   52  102    3    2    1    2  yes                 c part  4
     part   103    3  105    0    2    1    4  yes                 c part  5
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Glass liner                                                            *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld 106 lpil   100  101 lvc -78.0000    1.2500 lpil    35   36
ld 107 lpil   100  101 lvc -75.0000    1.2500 lpil    23   24 lpil    34   35
ld 108 lpil   100  101 lvc -60.0000    1.5000 lpil    22   23
ld 109 lpil    71  100 lvc   0.0000    1.6400
ld 110 lp 1  0.7003   10.7503 lpil     2   15
ld 111 lp 1  0.7003   10.7503 lpil     5    6 lpil     2   52
       lvc 180.0000    0.6000
ld 112 lod 1 0.0000 lod 2 0.0000
ld 113 lod 4 0.0000 lod 5 0.0000
ld 114 lod 3 0.0000 lpil     2   52
     part   100    1  106    4    3    2    4  yes                 c part  6
     part   106    1  107    4    3    2    1  yes                 c part  7
     part   107    1  108    4    3    2    5  yes                 c part  8
     part   108    1  101    4    3    2   20  yes                 c part  9
     part   101    1  109    4    3    2   60  yes                 c part 10
     part   109    1  102    4    3    2   30  yes                 c part 11
     part   102  112  110  113    3    2   10  yes                 c part 12
     part   110    2  111    5    3    2    5  yes                 c part 13
     part   111  114  104    6    3    2    4  yes                 c part 14
     part   104    3  103    6    3    2    4  yes                 c part 15
     part   103    3  102    6    3    2    2  yes                 c part 16
     part   102    3  109    6    3    2   30  yes                 c part 17
     part   109    3  101    6    3    2   60  yes                 c part 18
```

```
     part   101    3  100    6    3    2   30  yes                  c part 19
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Air Pocket                                                             *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld 115 lpil    35   36 lvc 180.0000    0.4
ld 116 lpil    33   34 lpil    36   37 lvc 180.0000    0.4
ld 117 lpil   100  116 lpil    36   37 lpil    37   38
     part   100  115  106    1    4    2    4  yes                  c part 20
t13  part     1  106   35  107    4    1    2  yes                  c part 21
     part   116   36  115  100    4    4    4  yes                  c part 22
     part    48   38 -117  100    4    9    9    4  yes             c part 23
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Glass liner support                                                    *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld 118 lpil    37   38 lpil    32   33
ld 119 lpil    46   47 lvc 180.0000    0.1
ld 120 lpil    31   32 lpil    32  119 lpil    32   33
     part    31 -120  118   38    5    3    9    1  yes             c part 24
     part   118   33  116   37    5    3    5  yes                  c part 25
     part   116   34   35   36    5    3    4  yes                  c part 26
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Tip protector                                                          *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld 121 lpil    23   24 lpil   34   35 lpil    21   24
     part    21   22   23 -121    6    5    6    2  yes             c part 27
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Air space                                                              *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld 122 lpil   100  101 lvc -54.0000    1.5000 lpil    41   42
ld 123 lpil    21   22 lpil    45   46
ld 124 lpil    46   47 lvc 117.0000    0.7500
ld 125 lpil   100  101 lvc -36.0000    1.5000 lpil    45   46
ld 126 lpil   100  101 lvc -27.0000    1.7500
ld 127 lpil   100  101 lvc -12.0000    1.6000
t13  part     1   22  123  122    7    2    6  yes                  c part 28
     part   123   32  119  124    7    1    8  yes                  c part 29
     part   123  124   46  122    7    5    8  yes                  c part 30
     part    47  119   32    0    7    2    1  yes                  c part 31
     part   122   46  123  123    7    5    3  yes                  c part 32
     part     1  122  123  125    7    6    6  yes                  c part 33
     part   126    1  125   16    7    6    3  yes                  c part 34
t13  part   127    1  126   16    7    2    5  yes                  c part 35
     part   127   16    1    0    7    2    4  yes                  c part 36
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Jacket Bottom                                                          *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld 128 lpil    31   38 lvc -90.0000    0.2000
ld 129 lpil    46   47 lpil    47   48 lvc -90.0000    0.2000
ld 130 lpil    42   43 lpil    43   44 lvc 200.0000    0.2000
     part   100   41  128   48    8    2    9  yes                  c part 37
     part   128   41  129   48    8    2    3  yes                  c part 38
     part  -129   41  122   46    8    4    5    2  yes             c part 39
     part   122   42 -130   46    8    4    5    2  yes             c part 40
     part   130   44   45   46    8    2    3  yes                  c part 41
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Jacket                                                                 *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld 131 lpil    45   46 lpil    44   45 lvc   0.0000    0.2000
ld 132 lpil    16  126 lvc   0.0000    0.2000
```

```
ld 133 lpil    16  127 lvc   0.0000    0.2000
ld 134 lpil    61   62 lpil    61   65 lvc 188.0000    0.2500
ld 135 lpil   102   16 lpil    14   15 lvc   0.0000    0.2500
     part    43   11  131   18    9    2    3  yes                 c part 42
     part  -131   11  132   16    9    4    3    2  yes            c part 43
     part   132   11  133   16    9    4    5  yes                 c part 44
     part   133   11  101   16    9    4    4  yes                 c part 45
     part   101   11  109   16    9    4   60  yes                 c part 46
t13  part   109   11 -134   16    9    4   22    1  yes            c part 47
     part   134   13 -135   16    9    9    8    8  yes            c part 48
     part    15   16  135  135    9    6    4  yes                 c part 49
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Air space                                                              *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
ld 136 lpil    16  102 lvc  21.5000    0.6000 lvc   0.0000    0.2500
ld 137 lpil    16  102 lvc  70.0000    0.7100 lvc   0.0000    0.5000
ld 138 lpil    52   53 lvc   0.0000    1.0000
ld 139 lpil    15  137 lvc 103.3000    0.7500
ld 140 lpil    15  110 lvc  78.6000    0.5000
ld 141 lvc   0.7003   10.7503  128.2500    0.1247
       lvc 180.0000    0.6500
ld 142 lpil     2  141 lvc  77.8000    0.5000
ld 143 lpil    52   53 lpil    64   65 lpil    62   63
t13  part    14   65  136   15   10    1    1  yes                 c part 50
t13  part   136   65  137   15   10    3    5  yes                 c part 51
     part   137   65  138  139   10    9    9  yes                 c part 52
     part    15  139  138  140   10    4    9  yes                 c part 53
     part     2  140  138  142   10    3    9  yes                 c part 54
     part   141  142  138   52   10    2    9  yes                 c part 55
     part   141   52    2    0   10    2    2  yes                 c part 56
     part   143  138   65   65   10    9    9  yes                 c part 57
     part    53  143   64  100   10   27   18  yes                 c part 58
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Cork stopper                                                           *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
     part    51   52  103  100   11   27    2  yes                 c part 59
     part   103   52  104  100   11   27    4  yes                 c part 60
     part   104   52  111  100   11   27    4  yes                 c part 61
     part   111   52  141  100   11   27    2  yes                 c part 62
     part   141   52   53  100   11   27    9  yes                 c part 63
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Cup                                                                    *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
     part   134   62  135   65   12    3    8  yes                 c part 64
     part   135   62  136   65   12    3    1  yes                 c part 65
     part   136   62  137   65   12    3    5  yes                 c part 66
     part   137   62  138   65   12    3    9  yes                 c part 67
     part   138   62  143   65   12    3   18  yes                 c part 68
     part   143   63  100   64   12    3   27  yes                 c part 69

assm

c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Merge subparts to form parts                                          *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
mgm     -1    2                                    c Fluid               Mat  1
mgm     -3    5                                    c Air over Fluid      Mat  2
mgm     -6   19                                    c Glass liner         Mat  3
mgm    -20   23                                    c Air Pocket          Mat  4
```

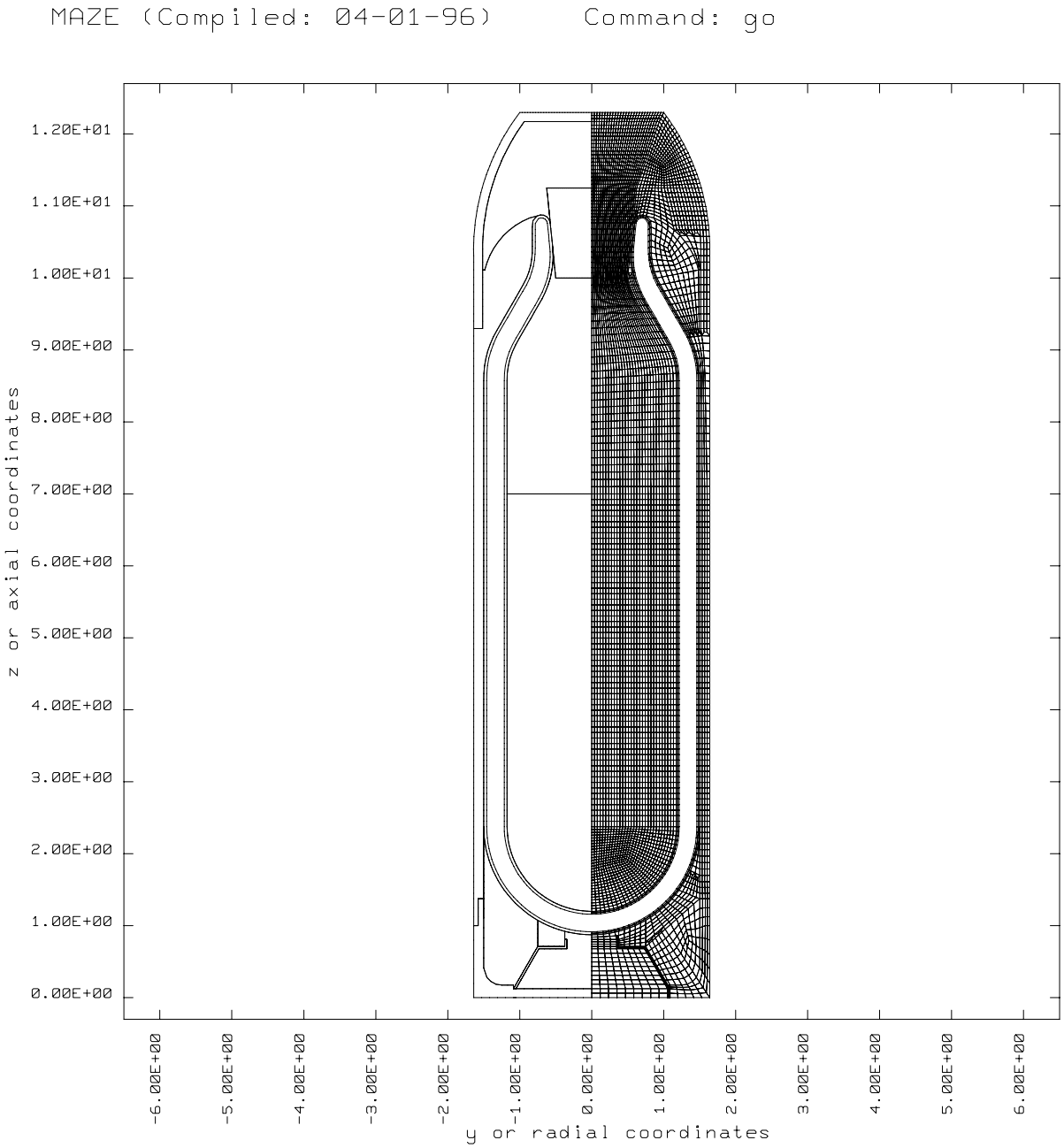
```
mgm    -24   26                                  c Glass liner support   Mat  5
                                                 c Tip protector         Mat  6
mg      29   30 p 29 b ess 3
mgm    -28   36                                  c Air Space             Mat  7
mgm    -37   41                                  c Jacket Bottom         Mat  8
mgm    -42   49                                  c Jacket                Mat  9
mgm    -50   56 p 50 b ess 3
mgm    -50   58                                  c Air Gap               Mat 10
mgm    -59   63                                  c Cork Stopper          Mat 11
mgm    -64   69                                  c Cup                   Mat 12

c Move nodes at corner of parts which will utilize an interface resistance, to avoid
c merging them.
pntr 1 0.01 7.59586811E-01 1.08599997E+01 p 6 b cbnr 1 0.76 10.85
pntr 2 0.01 1.50000000E+00 2.37500000E+00 p 6 b cbnr 2 1.49 2.375

c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c *  Merge parts                                                                  *
c * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
mg       1    3     c   31 nodes merged from parts    1 and    3
mg       1    6     c   91 nodes merged from parts    1 and    6
mg       3    6     c   36 nodes merged from parts    3 and    6
mg       3   59     c   34 nodes merged from parts    3 and   59
mg       6   20     c    6 nodes merged from parts    6 and   20
mg       6   27     c    6 nodes merged from parts    6 and   27
mg       6   28     c   20 nodes merged from parts    6 and   28
gm       6   50     c    5 nodes merged from parts    6 and   50
mg       6   59     c    4 nodes merged from parts    6 and   59
mg      20   24     c   22 nodes merged from parts   20 and   24
mg      20   27     c    2 nodes merged from parts   20 and   27
mg      20   37     c   10 nodes merged from parts   20 and   37
mg      24   27     c    9 nodes merged from parts   24 and   27
mg      24   28     c   10 nodes merged from parts   24 and   28
mg      24   37     c    3 nodes merged from parts   24 and   37
mg      27   28     c    5 nodes merged from parts   27 and   28
mg      28   42     c   13 nodes merged from parts   28 and   42
gm      28   37     c   15 nodes merged from parts   28 and   37
mg      37   42     c    7 nodes merged from parts   37 and   42
gm      42   50     c   12 nodes merged from parts   42 and   50
mg      42   64     c   12 nodes merged from parts   42 and   64
mg      50   59     c   38 nodes merged from parts   50 and   59
mg      50   64     c   60 nodes merged from parts   50 and   64

c return relocated nodes to original position following merges
pntr 3 0.01 0.76 10.85 p 6 b cbnr 3 7.59586811E-01 1.08599997E+01
pntr 4 0.01 1.49 2.375 p 6 b cbnr 4 1.50000000E+00 2.37500000E+00

c  * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c  enclosure radiation
c  * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

c Wavelength breakpoints for wavelength dependent emissivities
c An arbitrarily large wavelength is defined as a total emissivity is used
lambda  1  1.0E+10

c Emissivity (total hemispherical)
ecd  1  0.8 c plastic
ecd  2  0.8 c cork
ecd  3  0.2 c glass with reflective coating
```

```
ecd  4  0.9 c fluid
ecd  5  0.5 c steel
ecd  6  0.8 c rubber

c units in Kelvin and Steffan Boltzman in SI units
ercc  k  5.67E-08

c Cavity inside cup
pntr 10 .01 0.00000E+00 1.12500E+01
pntr 11 .01 5.76612E-01 1.07661E+01
pntr 12 .01 7.59587E-01 1.08600E+01
pntr 13 .01 1.51500E+00 1.01100E+01
pntr 14 .01 0.00000E+00 1.21730E+01
p 59 b bcrn p10 10 bcrn p11 11 ebc %p11 %p10 0 2
p  6 b bcrn p11 11 bcrn p12 12 ebc %p12 %p11 0 3
p 42 b bcrn p12 12 bcrn p13 13 ebc %p13 %p12 0 1
p 64 b bcrn p13 13 bcrn p14 14 ebc %p14 %p13 0 1

c Cavity inside of Liner
p 6 b ebcs 3 0 3

c Cavity between fluid, liner and cork
pntr 15 .01 0.00000E+00 7.00000E+00
pntr 16 .01 1.17500E+00 7.00000E+00
pntr 17 .01 5.43333E-01 1.04330E+01
pntr 18 .01 0.00000E+00 1.00000E+01
p  1 b bcrn p15 15 bcrn p16 16 ebc %p16 %p15 0 4
p  6 b bcrn p16 16 bcrn p17 17 ebc %p17 %p16 0 3
p 59 b bcrn p17 17 bcrn p18 18 ebc %p18 %p17 0 2

c Cavity in bottom
pntr 19 .01 1.08820E+00 1.25000E-01
pntr 20 .01 1.50000E+00 1.37500E+00
pntr 21 .01 1.50000E+00 2.37500E+00
pntr 22 .01 7.49974E-01 1.07601E+00
pntr 23 .01 7.50000E-01 7.10800E-01
p 37 b bcrn p19 19 bcrn p20 20 ebc %p20 %p19 0 1
p 42 b bcrn p20 20 bcrn p21 21 ebc %p21 %p20 0 1
p  6 b bcrn p21 21 bcrn p22 22 ebc %p22 %p21 0 3
p 27 b bcrn p22 22 bcrn p23 23 ebc %p23 %p22 0 6
p 24 b bcrn p19 19 bcrn p23 23 ebc %p19 %p23 0 5

c Cavity in bottom
pntr 24 .01  0.00000E+00 1.25000E-01
pntr 25 .005 1.05360E+00 1.25000E-01
pntr 26 .005 3.75000E-01 8.10800E-01
pntr 27 .01  3.74983E-01 9.22679E-01
pntr 28 .01  0.00000E+00 8.75000E-01
p 37 b bcrn p24 24 bcrn p25 25 ebc %p25 %p24 0 1
p 24 b bcrn p25 25 bcrn p26 26 ebc %p26 %p25 0 5
p 27 b bcrn p26 26 bcrn p27 27 ebc %p27 %p26 0 6
p  6 b bcrn p27 27 bcrn p28 28 ebc %p28 %p27 0 3

c  * * * * * * * * * * * * * * * * * * * * * * * * * * *
c  slide line to provide contact resistance
c  * * * * * * * * * * * * * * * * * * * * * * * * * * *
c Between Liner and Jacket
sln 1 7 500.0 0.0
slbp 6 42
```

```
c  * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c  external boundary conditions
c  * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c turbulent free convection external to Thermos
c ref: Holman, J.P., Heat Transfer, Fifth ed., pg 285.
p 37 b cbcs 2 0 [((70-32)/1.8)+273.15] [((70-32)/1.8)+273.15] 0 0.95 [1/3]
p 42 b cbcs 2 0 [((70-32)/1.8)+273.15] [((70-32)/1.8)+273.15] 0 0.95 [1/3]
p 64 b cbcs 2 0 [((70-32)/1.8)+273.15] [((70-32)/1.8)+273.15] 0 0.95 [1/3]
       cbcs 3 0 [((70-32)/1.8)+273.15] [((70-32)/1.8)+273.15] 0 1.43 [1/3]

c radiation external to Thermos
p 37 b rbcs 2 0 [((70-32)/1.8)+273.15] [((70-32)/1.8)+273.15] 0 [0.8*5.67E-08]
p 42 b rbcs 2 0 [((70-32)/1.8)+273.15] [((70-32)/1.8)+273.15] 0 [0.8*5.67E-08]
p 64 b rbcs 2 0 [((70-32)/1.8)+273.15] [((70-32)/1.8)+273.15] 0 [0.8*5.67E-08]
       rbcs 3 0 [((70-32)/1.8)+273.15] [((70-32)/1.8)+273.15] 0 [0.8*5.67E-08]


title
Thermos
alpha 1.0   c fully implicit
bwmo 1      c minimize bandwidth
rtyp 1      c enclosure radiation using view factors (diffuse reflectors)
anal 2      c transient analysis with lumped mass matrix
step 0      c fixed time step
delt 30     c time step size
prti 2      c printing interval
plti 2      c plotting interval
sbrf 600    c steps between restart file
start 0     c initial time
term 14400  c termination time
nonl 1      c nonlinear, mtl properties evaluated at gauss point temperature
wbcd topaz2d
fson go !!  c Verify that parts were merged
t0 [((70-32)/1.8)+273.15]      c initial temperature of all nodes
mrit 1 [((180-32)/1.8)+273.15] c initial temperature of nodes in fluid
csf 0.0254  c convert units from inch to meter

c  * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
c  material properties: meter, joule, second, kelvin, kg
c
c       density:       kg/m^3
c       heat gen:      W/m^3
c       Cp:            J/kg-K
c       conductivity:  W/m-K
c  * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

tmat 1
water
mt 3
den 1000.
temp 3 273.15  323.15  373.15
cp 3   4217.47 4174.38 4209.104
con1 3 550614  .646428 .6828288

tmat 2
air
mt 3
den 1.29
```

```
temp 5 78.7999 123.15  173.15  273.15  373.15
cp 5   1004.16 1004.16 1001.65 996.629 991.608
con1 5 .006904 .011053 .015732 .024142 .031798

tmat 3
glass
mt 3
den 2220.
temp 4 123.15 173.15 273.15 373.15
cp 4   364.01 493.71 681.99 853.54
con1 4 .87864 .87864 1.0878 1.1994

tmat 4
air
mt 3
den 1.29
temp 5 78.7999 123.15  173.15  273.15  373.15
cp 5   1004.16 1004.16 1001.65 996.629 991.608
con1 5 .006904 .011053 .015732 .024142 .031798

tmat 5
steel
mt 3
den 7860.00
temp 4 273.15  298.15  348.15  473.15
cp 4   439.32  460.24  502.08  564.84
con1 4 73.7964 71.1280 68.4286 61.6803

tmat 6
rubber
mt 1
den 900.0001
cp 1   1966.48
con1 1 .087864

tmat 7
air
mt 3
den 1.29
temp 5 78.7999 123.15  173.15  273.15  373.15
cp 5   1004.16 1004.16 1001.65 996.629 991.608
con1 5 .006904 .011053 .015732 .024142 .031798

tmat 8
plastic
mt 1
den 1800.
cp 1   1255.2
con1 1 .2092

tmat 9
plastic
mt 1
den 1800.
cp 1   1255.2
con1 1 .2092

tmat 10
air
```

```
mt 3
den 1.29
temp 5 78.7999 123.15  173.15  273.15  373.15
cp 5   1004.16 1004.16 1001.65 996.629 991.608
con1 5 .006904 .011053 .015732 .024142 .031798

tmat 11
cork
mt 1
den 130.
cp 1   2301.2
con1 1 .050208

tmat 12
plastic
mt 1
den 1800.
cp 1   1255.2
con1 1 .2092

end
```


MAZE (Compiled: 04-01-96) Command: go
z or axial coordinates
1.20E+01
1.10E+01
1.00E+01
9.00E+00
8.00E+00
7.00E+00
6.00E+00
5.00E+00
4.00E+00
3.00E+00
2.00E+00
1.00E+00
0.00E+00
-6.00E+00
-5.00E+00
-4.00E+00
-3.00E+00
-2.00E+00
-1.00E+00
0.00E+00
1.00E+00
2.00E+00
3.00E+00
4.00E+00
5.00E+00
6.00E+00
y or radial coordinates

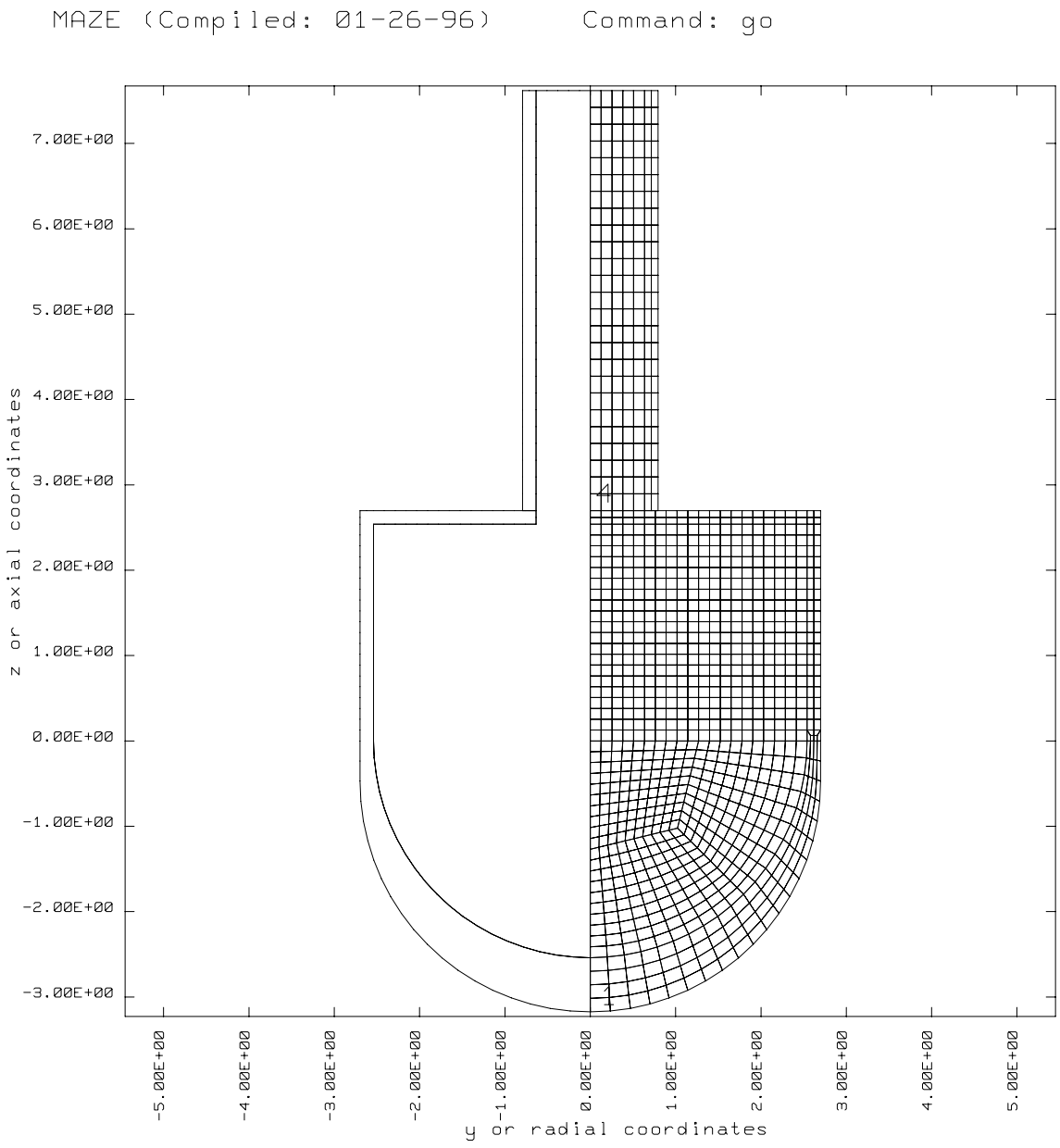
9.4 CHEMICAL TOPAZ2D: Calorimeter Analysis

```
1
c
c Calorimeter Analysis
c
c Dimensions:  cm/gm/sec/K
c
c
c Define Lines
c
maztl 0.001
lpoff ppoff
c
ld 1 lp 2 0 2.54 5.54 2.54
ld 2 lp 2 2.54 -3 2.54 5.54
lcc 1 0 0 -90.0 0 2.54
ld 4 lp 1 0.0 -3.175
lpt 2.69875 -3.0 2.69875 1.0 2.69875
ld 5 lp 2 0 0 2.69876 0
ld 6 lp 2 0 -5 0 7.62
ld 7 lp 2 0.635 2.54 0.635 7.62
ld 8 lp 2 0 7.62 3.635 7.62
ld 9 lp 2 0.79375 2.54 0.79375 7.62
ld 10 lp 2 0 2.69875 5 2.69875
ld 11 lp 2 2.69875 -3 2.69875 2.7
c
c Define Parts
c
part 10 7 8 6 4 5 25 yes c Part 1 sst
part 5 2 1 6 4 20 20 yes c Part 2 sst
part 6 3 5 5 4 10 10 yes c Part 3 sst
part 1 7 10 6 4 5 2 yes c Part 4 sst
part 4 5 3 6 1 20 4 ye sc Part 5 al2o3
t12 part 1 2 5 11 1 2 20 yes c Part 6 al2o3
part 1 11 10 2 1 2 2 yes c Part 7 al2o3
part 1 2 10 7 1 15 2 yes c Part 8 al2o3
part 10 9 8 7 1 2 25 yes c Part 9 al2o3
c
c Assemble and Merge Parts
c
assm
```

```
m 1 4 m 4 2 m 2 3 m 3 5 m 5 6 m 6 7 m 7 8 m 2 6 m 2 8 m 8 4 m 9 1
c
c Define initial temperature and boundary conditions
c
lcd 1 4
0.0 298.2
10.0 2575.0
100.0 2975.0
200 1375.0
c
lcd 2 2
0.0 1.354e-12
20000.0 1.354e-12
c
c Radiation on surface of al2o3
c
p 9 b rbcs 2 1 1.0 1.0 2 0.3
p 8 b rbcs 3 1 1.0 1.0 2 0.3
p 7 b rbcs 2 1 1.0 1.0 2 0.3 rbcs 3 1 1.0 1.0 2 0.3
p 6 b rbcs 2 1 1.0 1.0 2 0.3
p 5 b rbcs 2 1 1.0 1.0 2 0.3
c
c Enclosure radiation
c
c ercc k 1.345e-12
c lambda 1 1.0e+10
c ecd 1 0.2
c ecd 2 0.8
c tic 1 1126 298.15
c
c Define slidelines
c
sln 1 7 1.0 0.0 p 9 b msrs 1 p 8 b slv 1035 1049
c
c Set title, control data, and timing
c
title
cal3a - Calorimeter thermal test simulation
c
alpha 0.5 c time integration parameter 1.0-implicit 0.5-CN
c - plane geometry default-axisymmetric plane-plane geometry
bwmo 1 c bandwidth minimization 1-on 0-off
flux 1 c node heat flux calculation 0-off 1-on
phch 1 c phase change calculation 0-of 1-on
nsmd 0 c solution method 0-fizzle CRAY 1-actol SUN
c radiation calculation type 1-view factor 0-ex factor
```

```
analysis 2
step 1 c time step mode 0-fixed 1-variable
iprt 5000 c steps between print output
iplt 300 c steps between plot output
sbrf -100 c number of time steps between restart files
start 0.0 c initial problem time
term 110.0 c termination time
delt 1.0 c time step if fixed - initial variable TS
dtmin 0.001 c minimum time step - set for variable TS
dtmax 10.0 c maximum time step - set for variable TS
tmpmax 10.0 c maximum temperature change in TS - set for variable TS
tssf 0.5 c time step scaling factor - set for variable TS
nonl 1 c type of problem 0-linear 1-nonlinear
nbsr 1 c number of time steps between stiff matrix ref.
c nbei - number of time steps between equilibrium iterations
msrf 10 c maximum number of reformation per time step
nibsr 1 c max. number of equilibrium iterations per cond. matrix ref.
tctol 0.001 c convergence tolerance for equilibrium iterations def=0.001
c relaxation parameter - def (1.0-ss 0.05 tran)
c
c Chemistry Control
c
rprt 5000.0 c time between printing
rplt 30.0 c time between plotting
nrx 1 c number of reactions
nmix 1 c number of mixture materials
c
mixn 5 5 6 7 8 9 c parts composed of reacting material
eubg 2 c coming from material for effective eulerian
eued 6 c going to material for effective eulerian
eual 0.05 c error tolerance for effective eulerian
euar 3.8485 c effective eulerian area
c
c Create output
c
wbcd topaz2d
c
c Define Materials
c
c Baseline temperature of all nodes
c
t0 298.0
tmat 1
al2o3 MIX
mt -8 den 5.00e-01
mixs 2 2 1.0
```

```
tmat 2
trans aloxf
mt 105 den 5.00e-01
tcp5 11
0.000e-01 0.000e-01 7.315e+01 9.560e-03 2.231e+02 1.400e-01 2.731e+02 1.740e-01
3.732e+02 2.220e-01 4.731e+02 2.450e-01 5.731e+02 2.651e-01 7.731e+02 2.801e-01
1.173e+03 3.107e-01 1.973e+03 3.119e-01 1.973e+04 3.119e-01
tcn5 2
2.731e+02 9.560e-01 3.732e+03 9.560e-01
tref 273.15 q0 0.0
tmat 3
aloxf
mt 105 den 5.000e-01
tcp5 11
0.000e-01 0.000e-01 7.315e+01 9.560e-03 2.231e+02 1.400e-01 2.731e+02 1.740e-01
3.732e+02 2.220e-01 4.731e+02 2.450e-01 5.731e+02 2.650e-01 7.731e+02 2.801e-01
1.173e+03 3.107e-01 1.973e+03 3.119e-01 1.973e+04 3.12e-01
tcn5 4
6.231e+02 9.560e-05 1.073e+03 4.199e-04 1.273e+03 7.50e-04 1.273e+04 7.500e-04
tref 273.15 q0 0.0
tmat 4
304 sst
mt 105 den 7.920e+00
tcp5 2
2.731e+02 1.200e-01 6.731e+02 1.350e-01
tcn5 6
7.315e+01 1.650e-02 1.732e+02 2.600e-02 2.731e+02 3.200e-02 3.732e-02 3.901e-02
7.731e+02 5.201e-02 9.731e+02 6.200e-02
tref 273.15 q0 0.0
c
c Define Chemistry Control
c
gass 1.0 errx 0.0001 itrx 30 mint 400.0 maxt 1000.0
c
c mpl1 3
c mpl2 9
c
c Define Reactions
c
reac
c typ 3 rkk -4.09 eex 0.0 stos 2 3 -1.0 1.0 iprs 2 3 1 0
end
```


MAZE (Compiled: 01-26-96) Command: go
z or axial coordinates
7.00E+00
6.00E+00
5.00E+00
4.00E+00
3.00E+00
2.00E+00
1.00E+00
0.00E+00
-1.00E+00
-2.00E+00
-3.00E+00
-5.00E+00
-4.00E+00
-3.00E+00
-2.00E+00
-1.00E+00
0.00E+00
1.00E+00
2.00E+00
3.00E+00
4.00E+00
5.00E+00
y or radial coordinates

10 COMMAND DEFINITIONS: QUICK REFERENCE GUIDE

General Commands (p. 17)

!	Suspend indicator.
{ ... } C	Comment delimiters.
CFILE	Return interactive control of MAZE to command file.
DYNA2D / NIKE2D	Set MAZE analysis preference flag.
END / T	End / Terminate MAZE.
FLDID / NOFLDID	Include / Omit verbose descriptors in MAZE output file.
MAZTL *tolerance*	Establish MAZE tolerance specification.
PARAMETER p_1 [e_1] ... p_n [e_n];	Assign value of arithmetic expression *e* to parameter *p*.
QUIT	Exit MAZE. No MAZE output is generated.
SHOW p_1 ... p_n;	Display current values assigned to parameters.
TRAP	Terminate MAZE upon occurrence of serious errors.
TTY	Return controlof MAZE to computer terminal.
TV	Select new graphics output device.

Graphics Commands (p. 19)

FRAME / NOFRAME	Enable / disable display of reference axes and tick marks.
GRID / NOGRID	Enable / disable display of grid lines.
GSET *r z* δ	Center display at coordinates (*r,z*).
Z *r z* δ	Zoom to coordinates (*r,z*) using window δ.

Phase I

Point Creation (p. 20)

FLPIL l_1 l_2 *r_variable z_variable*	Define point at intersection of lines.
PTD *symbol r z*	Establish *symbol* representing coordinate pair (*r,z*).
PTSV	Display point definitions.

Line Graphics (p. 21)

LNON / LNOFF	Enable / disable display of line numbers.
LPON / LPOFF	Enable / disable line plotting commands.
LV	Display all lines.
LVI *n* l_1 ... l_n	Display *n* lines consisting of line numbers l_1 ... l_n.
LVS l_1 l_2	Display all lines between numbers l_1 and l_2, inclusive.
LZOOM l_1	Center current display on line number l_1.

Line Segment Definitions (p. 21)

LD n — Begin definition of line.

Straight Lines (p. 21)

LP $n\ r_1\ z_1 \ldots r_n\ z_n$ — Define points to be added to current line definition.
LPIL $l_1\ l_2$ — Define point for current line at intersection of lines.
LRL $n\ r_c\ z_c\ l\ \Theta_1 \ldots \Theta_n$ — Define radial lines.
LVC $\Theta\ l$ — Define line segment by vector of length.
LVC $r_1\ z_1\ \Theta\ l$
LVC $r_2\ z_2\ \Theta\ -l$

Curved Lines (p. 22)

CLAP $r_1\ z_1\ r_c\ z_c$ — Define circular arc.
CUBIC $\Theta_1\ r_2\ z_2\ \Theta_2$ — Define line segment using third-order cubic equation.
PCUBIC — Display parameters and constants of most recent line segment definition generated by CUBIC.

LAD $r_c\ z_c\ \Theta$ — Define circular arc.
LAP $r_1\ z_1\ r_c\ z_c$ — Define circular arc.
LAR $r\ z\ R$ — Define circular arc.
LAT $r_1\ z_1\ r_2\ z_2\ R$ — Define circular arc.
LCC $n\ r_c\ z_c\ \Theta_1\ \Theta_2\ r_1 \ldots r_n$ — Define lines consisting of circular arcs.
LEP $a\ b\ r_c\ z_c\ \Theta_1\ \Theta_2\ \Phi$ — Define elliptic arc.
LPT $r_1\ z_1\ r_2\ z_2\ R$ — Define circular arc.
LPTA $r_c\ z_c\ R$ — Define line segment beginning at last point defined and terminating at its tangency point on arc.

LTAS $r_{c1}\ z_{c1}\ rot\ r_{c2}\ z_{c2}\ R_2$ — Define line segment consisting of circular arc followed by straight line segment.

LTP $r\ z\ R$ — Define circular arc.
ML $l_1\ l_2$ — Append lines.

Copied / Offset Lines (p. 24)

LO $l\ r_1\ z_1\ r_2\ z_2$ — Define line segment by offsetting a segment of line.
LOD $l\ \delta$ — Define line segment by offsetting line segment.
LSTL $l\ \Delta r\ \Delta z$ — Define line segment by translating entire line.
LT $l\ \Delta r\ \Delta z$ — Translate line.
LTM $n\ l_1 \ldots l_n\ \Delta r\ \Delta z$ — Translate lines.
LTS $l_a\ l_b\ \Delta r\ \Delta z$ — Translate consecutive lines.
VLOD $l\ \delta_1\ \delta_2$ — Define line segment by offsetting from line.

Tab Cell Data (p. 25)

LTBC $n\ \Theta\ \Delta\Theta\ S\ R_1 \ldots R_n$ — Define line segment with tab cell data.
LTBO $m_1\ \delta_1 \ldots m_k\ \delta_k$ — Define line segment by offsetting last line segment defined with commands LTBC or LTBO.

Auxiliary Line Operations (p. 25)

CKL $l_1\ l_2$ — Remove external angles ≥ 120°; coalesce duplicated points.
DELETE l — Delete line.
LPRI l — Print coordinates of line on terminal.
MLN — Print maximum line number used.
NDL — Print numbers of all lines deleted.

Part Graphics (p. 26)

LVPV	Display all lines and parts.
PNON / PNOFF	Enable / disable display of part numbers within plots.
PPON / PPOFF	Enable / disable display of part plotting.
PV	Display all parts.
PVI $n\ p_1 \ldots p_n$	Display parts.

Part Definitions (p. 27)

PART $L_1\ L_2\ L_3\ L_4$ *material k m*	Define four sided region to be a part.
QUAD $r_1\ z_1$ ... $r_4\ z_4$ *material k m*	Define four sided region to be a part.
RECT $r_1\ z_1\ r_3\ z_3$ *material k m*	Define the rectangular region to be a part.
T12	Transition elements by two along side L_3.
T13	Transition elements by three along side L_3.
T21	Transition elements by one-half along side L_3.
T31	Transition elements by one-third along side L_3.
TRANS	Change element distribution of part.
PART $L_1\ L_2\ L_3\ L_3$ *material k m*	Define three sided region to be part.
PART $L_1\ L_2\ L_3\ 0$ *material k m*	Define three sided region to be part.
TRIQ $r_1\ z_1\ r_2\ z_2\ r_3\ z_3$ *material k m*	Define three sided region to be part.
TRIT $r_1\ z_1\ r_2\ z_2\ r_3\ z_3$ *material k m*	Define three sided region to be part.
PART $L_1\ L_2\ L_2\ L_2$ *material k m*	Define region bounded by line and arc to be part.
PART $L_1\ L_1\ L_1\ L_1$ *material k m*	Define region bounded by elliptic arc to be part.
PART ... / QUAD ... / RECT ... { *material -k m* R_1 / *material k -m* R_2 / *material -k -m* $R_1\ R_2$	Define four sided part with nodal spacing and element sizing that transitions smoothly across part.
PART ... / QUAD... / RECT... *-material k m* R_1 ... R_4	Define four sided part with independent nodal spacing and element sizing that transitions smoothly across part.
PART $L_1\ L_2\ L_3\ L_4$ { *material 0 m* / *material k 0* / *material 0 0*	Define four sided part using line points to define nodal locations.
PART $-L_aL_bL_cL_d$ *mt k m* $n^a{}_1\ n^a{}_{pa-2}$	Define four sided part using line points to define nodal locations.

Nodal Spacing Based on Angular Position (p. 30)

AZON $n\ S_1 \ldots S_n\ r_c\ z_c$ / AZOFF	Enable / Disable equal angular spacing of nodes.

Part Duplications (p. 30)

CLONE *n material* $\Delta r\ \Delta z\ \Theta$	Define part by duplicating part *n*.
RFLIP *n material*	Duplicating part and rotate about *r*-axis.
ZFLIP *n material*	Duplicating part and rotate about *z*-axis.

Nodal Spacing Based on Angular Position (p. 30)

AZON $n\ S_1 \ldots S_n\ r_c\ z_c$ / AZOFF	Enable / Disable equal angular spacing of nodes.

Auxiliary Part Commands (p. 31)

AOR Θ	Establish node at vertices $< \Theta^\circ$ in boundary lines.
BPN *n*	Number parts consecutively beginning with *n*.
DP *m*	Delete part.
FIXP *n*	Set *r*-and *z*-constraints for part.
GEOZ	Switch between algebraic and geometric zoning.
MG *n m*	Merge interface nodes of parts having same coordinates.
NLD *L m*	Establish node line definition.
REXT *n rx*	Scale part in *r* direction.
RMIN *n rmin*	Translate part to have minimum r-coordinate value.
ZEXT *n zx*	Scale part in *z* direction.
ZMIN *n zmin*	Translate part to have minimum z-coordinate value.

Regions (p. 32)

ARCR *region rad a b* r_c z_c Θ_1 Θ_2 ϕ	Define arc *region*.
BCRN *symbol region*	Define node number contained in *region* to *symbol*.
CRVR *region tol n* r_1 z_1 ... r_n z_n	Define curve *region*.
LDR *region tol line*	Define line *region*.
LINR *region tol* r_1 z_1 r_2 z_2	Define linear *region*.
PNTR *region rad r z*	Define point *region*.
RV	Display all previously defined regions.
RVI *n* $region_1$ ... $region_n$	Display previously defined regions.

Transition From Phase I To Phase II

Mesh Assembly (p. 34)

ASSM	Assemble mesh from all previously defined parts.
PASSM *n* p_1 … p_n	Assemble mesh from subset of parts.

Phase II

General Commands (p. 35)

B	Establish boundary nodes defining sides of part.
BLEND *option*	Set smoothing option.
CBNR *region r z*	Change coordinates of single boundary node.
CN *m r z*	Assign node new coordinates.
CNMN *m n*	Modify coordinates of node.
FLCD *id* t_0 t_n *n [function(*t*)]*	Define load curve containing time-function points.
FLIP	Interchange axes of symmetry.
GS	Smooth all parts.
LCD *id m* t_1 f_1 ... t_m f_m	Define load curve.
P *n*	Part for modification or boundary node determination.
R	Restore mesh.
S	Smooth mesh of part.
SIDE	Establish boundary defining sides of part.

Graphics Commands (p. 36)

A	Display all slidelines.
AML	Display all master sides of slidelines.
AS *m n*	Display slidelines.
ASL	Display all slave sides of slidelines.
CNPO / CNPS	Display / do NOT display corner nodes.
DBN	Delete boundary nodes from boundary plots.
DSN	Delete side numbers from boundary plots.
ELPLT	Display element numbers on mesh .
ELPM $n\ p_1 \ldots p_n$	Plot element numbers on mesh of parts.
G	Display complete mesh grid with part numbers.
NDPLT	Display node numbers on mesh of materials.
NDPM $n\ p_1 \ldots p_n$	Plot node numbers on mesh of parts.
LCV	Display all load curves.
LCVI $n\ lc_1 \ldots lc_n$	Display load curves.
O	Display outline of parts with part numbers.
OG	Display outline of parts with part numbers over grid.
TE $r\ z\ \Delta l$	Display element numbers and centroid coordinates.
TN $r\ z\ \Delta l$	Display node numbers and coordinates of nodes.
TNC *n*	Display nodal coordinates.

Merging (p. 37)

GM $p_n\ p_m$	Merge common interface nodes of parts.
M $p_n\ p_m$	Merge common interface nodes of parts having same coordinates.
MG $p_n\ p_m$	Merge common interface nodes of parts having same coordinates. Parts with same material will be merged.
MGM $m\ p_0\ p_1 \ldots p_m$	Merge parts to form new part.
MGM $-p_0\ p_m$	Alternative form of MGM.
MGN *n m*	Merge nodes.

Nodal Modification and Spacing (p. 38)

BD *m n* BDS *s*	Remove kinks from boundary.
EA *m n* EAS *s*	Assign spacing of boundary nodes.
ER *m n* ERS *s*	Assign equal spacing of boundary nodes in *r* direction.
ES *m n* ESS *s*	Assign equal spacing of boundary nodes.
EZ *m n* EZS *s*	Assign equal spacing of boundary nodes in *z* direction.
VA *m n ratio* VAS *s ratio*	Assign spacing of boundary nodes.
VS *m n ratio* VSS *s ratio*	Assign variable spacing of boundary nodes.

Nodal Boundary Conditions: DYNA2D - NIKE2D (p. 38)

NBC *m n code* NBCR *r code* NBCS *s code*	Assign boundary condition code to nodes.
NBCC *corner code*	Define boundary constraint on *corner.*
RCON *R*	Constrain in horizontal direction nodes on line $r = R$.
ZCON *Z*	Constrain in vertical direction nodes on line $z = Z$.

Nodal Loads: DYNA2D - NIKE2D (p. 39)

CNL *m n k* r_1 r_2 *i* CNLC *corner_node k* r_1 r_2 *i* CNLS *s k* r_1 r_2 *i*	Assign concentrated nodal load direction.
PBC *m n k* r_1 r_2 PBCR *r k* r_1 r_2 PBCS *s k* r_1 r_2	Assign pressure loads boundary condition.
SBC *m n k sf pf* r_c z_c Θ *radius s* SBCS *side k sf pf* r_c z_c Θ *radius s*	Apply spatially nonlinear pressure boundary condition.

Prescribed Nodal Kinematics (p. 40)

DBC *m n k* r_1 r_2 *i* (NIKE2D) DBCR *r k* r_1 r_2 *i* DBCS *s k* r_1 r_2 *i*	Assign displacement time history direction.
IAV ω r_c z_c	Set initial angular velocity.
IV v_r v_z	Set initial velocity components of parts.
IVN *m n* v_{rm} v_{zm} v_{rn} v_{zn}	Set initial velocity components of nodes.
IVP *n* v_r v_z	Set initial velocity components of part.
NRBN *m n* (DYNA2D) NRBR *r* NRBS *s*	Assign non-reflecting boundary condition.
VBC *m n k* r_1 r_2 *i* (DYNA2D) VBCS *s k* r_1 r_2 *i*	Assign velocity time history direction.

Slideline Definitions (p. 41)

MSR *m n* MSRR *r* MSRS *s*	Define master side boundary.
SLBMP p_n p_m	Add slideline between merged parts.
SLBP p_n p_m	Add slidelines between adjacent parts.
SLN *n type* (DYNA2D, NIKE2D)	Define slideline *type.*
SLN *n* 4 *f* (DYNA2D, NIKE2D)	Define slideline of type 4: frictional sliding with voids.
SLN *n* 5 r_t z_t r_h z_h (DYNA2D)	Define slideline of type 5: stone wall.
SLN *n* 5 *f* ε^p_{break} (NIKE2D)	Define slideline of type 5: tie breaking.
SLN *n* 7 *q r* (TOPAZ)	Define slideline of type 7: thermal.
SLNA Θ_1° Θ_2° (DYNA2D)	Add slideline extensions to master surface of DYNA2D sliding only and frictionless sliding with void slidelines.
SLV *m n* SLVR *r* SLVS *s*	Define slave side boundary.
SLVM *material_number* (DYNA2D)	Set slave nodes for DYNA2D slideline type 5: stonewall.
SLVN *m n* (DYNA2D)	Set slave nodes for DYNA2D slideline type 5: stonewall.
SLVP *part_number* (DYNA2D)	Set for DYNA2D slideline type 5: stonewall.
SMNO *offset* (DYNA2D)	Add *offset* to all nodes specified with command SLVN.

Slideline Control (p. 43)

ATN *tolerance* (NIKE2D)	Set Lagrange augmentation tolerance in normal direction.
ATT *tolerance* (NIKE2D)	Set Lagrange augmentation tolerance in tangential direction.
LAF *flag* (NIKE2D)	Set Lagrange augmentation.
MSDF *flag* (NIKE2D)	Set master surface description.
SLFS $\varepsilon_{failure}$ (NIKE2D)	Set slideline failure strain.
SLNEXT on | off (DYNA2D)	Enable / disable slideline extension bypass option.
SLNI *n m* (DYNA2D)	Set slideline intersection.
SLNP *factor*	Assign slideline penalty function.
SLNS *tolerance*	Set tolerance for determining initial voids.
SPF *flag* (NIKE2D)	Set small penetration.
SSDF *flag* (NIKE2D)	Set slave surface description.

Explosives: DYNA2D (p. 44)

BDET *m n*	Assign detonation line boundary condition.
RDET *r*	
SDET *s*	
BLAST *option lut_id* v_1 v_2	Set blast detonation parameters.
DECAY α *n reference_distance*	Set decay multiplier parameters of load curve.
DETP *n* t_l *m*	Assign nodes to be located at detonation point.
DETC *side* t_l *m*	
GUN *option lut_id* $value_1$	Set gun firing parameters.
LDET *m* n_1 ... n_m	Establish detonation line.
LUT *id n* d_1 t_1 ... d_n t_n	Define lookup table containing distances and times for use in establishing pressure boundary curve parameters.
SDVEL *vos vis*	Establish detonation velocity of high explosive.
SHAD *m n*	Establish shadow boundary.
SSHADs	

Arbitrary Lagrangian-Eulerian Formulations: DYNA2D (p. 46)

ALE / ENDALE	Initiate / terminate ALE material formulation sequence.
ABS *begin end material type*	Set ALE boundary segment *type*.
ABSR *region type*	
ABSS *side type*	

Nodal Constraints: DYNA2D - NIKE2D (p. 47)

CNP *cnode m n i*	Establish constrained nodal pair direction.
CNPB *cnode s i*	

J-Integral: NIKE2D (p. 48)

JCOORD c_x c_z	Set components of crack tip location.
JCT *m n*	Include boundary nodes in crack tip definition.
JCTC *corner*	
JCTR *r*	
JCTS *side*	
JINT *contours* p_x p_z	Establishment of J-Integral mode.
JPHASE *plus minus larrot*	Set mode mixity separation option for homogeneous isotropic elastic or interfacial isotropic elastic cracks.
JTHERM *option*	Set J-Integral thermal *option*.

Boundary Conditions: TOPAZ (p. 49)

CBC *m n k* r_1 r_2 *j* r_3 r_4 CBCR *r k* r_1 r_2 *j* r_3 r_4 CBCS *s k* r_1 r_2 *j* r_3 r_4	Assign convection boundary condition.
FBC *m n k* r_1 r_2 FBCR *r k* r_1 r_2 FBCS *s k* r_1 r_2	Assign flux load boundary condition.
RBC *m n k* r_1 r_2 *j* r_3 RBCR *r k* r_1 r_2 *j* r_3 RBCS *s k* r_1 r_2 *j* r_3	Assign radiation boundary condition.
TBC *m n k r* TBCR *r k r* TBCS *s k r*	Assign temperature boundary condition.
T0 *temperature*	Assign nodal initial / reference temperature .

Element Heat Generation: TOPAZ (p. 50)

EGR *m n k r*	Assign element heat generation boundary condition.

Enclosure Radiation: TOPAZ (p. 50)

EBC *m n k l* r_1 EBCR *r k l* r_1 EBCS *s k l* r_1	Assign an enclosure radiation boundary condition.
ECD *n* e_1 ... e_n	Establish emissivity curve definition.
ERCC *u c r s*	Establish enclosure ratiation parameters.
LAMBDA *n* λ_1 ... λ_n	Establish wavelength breakpoints.

Miscellaneous Boundary Conditions (p. 51)

MBCS	Write miscellaneous boundary conditions on all sides of all parts to mazout file.

DYNA2D Control (p. 52)

TITLE	Define problem title in MAZE output file.
INPSD *ndmat numeld nummas*	Set discrete springs, dampers, and masses.
BFGR *k s*	Set body force load due to base acceleration in *r* direction
BFGZ *k s*	Set body force load due to base acceleration in *z* direction
BFSX *k s*	Set body force load due to angular velocity about *x*-axis.
BFSZ *k s*	Set body force load due to angular velocity about *z*-axis.
IAUTO *option n* m_1 ... m_n	Set automatic contact of materials.
SCS *scope*	Set *scope* of contact searching.
SFAS *frequency*	Set search *frequency* for automatic slidelines.
FASP *scale*	Set *scale* factor for automatic slideline penalty number.
LVFC μ_s	Set low velocity friction coefficient μ_s.
HVFC μ_k	Set high velocity friction coefficient μ_k.
EFDC β	Set exponential friction decay constant β.
TERM *time*	Terminate calculation at *time*.
ITSS Δt_0	Set initial time step size.
SBRF *steps*	Set number of time *steps* between restart dumps.

TSSF *scale*	Set *scale* factor for computed time step size.
RFMTS *factor*	Set reduction *factor* to determine minimum time step.
REZO *begin end between*	Set automatic re-zoning time parameters.
NPBK *m* n^1_1 n^1_2 ... n^m_1 n^m_2	Write nodal data for *m* blocks into MAZE output file.
EPBK *m* e^1_1 e^1_2 ... e^m_1 e^m_2	Write element data for *m* blocks into MAZE output file.
TED *option*	Set output *option* for internal energy.
SDO *option*	Set *option* for chemistry dump output.
HVDF *flag*	Set history variable dump *flag*.
PVDF *flag*	Set peak value dump *flag*.
TIBU Δt	Set time interval between updates of analysis display.
TSBS steps	Number of time *steps* between status updates print to file.
IGM *type*	Set geometry *type*.
BRODE	Initiate Brode function sequence.
ENDBRODE	Terminate Brode function sequence.
GRVS *a n* ρ_1 z_1 ... ρ_n z_n	Set gravity stress initialization.
TEO *option*	Set thermal effects *option*.
DHQT *method*	Hourglass stabilization *method*.
DHGQ Q_h	Hourglass viscosity coefficient.
DBQT *type*	Bulk viscosity *type*.
DQQ Q_q	Quadratic shock viscosity coefficient.
DQL Q_l	Linear shock viscosity coefficient.
SRDR *rate*	Stress *rate* default reset.

NIKE2D Control (p. 57)

TITLE	Define problem title in MAZE output file.
NPBK *m* n^1_1 n^1_2 ... n^m_1 n^m_2	Write nodal data for blocks into the MAZE output file.
EPBK *m* e^1_1 e^1_2 ... e^m_1 e^m_2	Write element data for blocks into the MAZE output file.
BFGR *k s*	Set body force load due to base acceleration in *r*-direction
BFGZ *k s*	Set body force load due to base acceleration in *z*-direction
BFSZ *k s*	Set body force load due to angular velocity about *z*-axis.
NCNM *n*	Set number of concentrated nodal masses.
NCND *n*	Set number of concentrated nodal dampers.
SMOPT *flag*	Set element formulation *flag*.
NEIP *flag*	Set integration order *flag*.
NBFL *n*	Set *n*umber of element body forces.
TEO *option*	Set thermal effects *option*.
ITCURV *load_curve*	Set *load_curve* number for temperature vs. time.
ITRF *flag*	Set initial temperature reference *flag*.
IEPD *flag*	Set element plot database *flag*.
IGM *type*	Set geometry *type*.
AF *flag*	Set analysis *flag*.
NEIG *n*	Perform eigenvalue analysis.
BWMO *flag*	Set bandwidth minimization *flag*.
IOOSF *flag*	Set out-of-core solution *flag*.
PCM *percent*	Set percentage of computer memory to be used.
SM *method*	Set solution *method* to be used.
NIP1 γ	Set Newmark parameter γ.
NIP2 β	Set Newmark parameter β.
TSSF *scale*	Set *scale* factor for computed time step size.

NIKE2D Solution Definitions (p. 60)

DELT $\Delta time$	Set time step size.
NSTEP *steps*	Set number of time *steps*.
PRTI *step_interval*	Set node and element dump *step interval* for printing.
PLTI *step_interval*	Set node and element dump *step interval* for ORION.
PRTT $\Delta time_{print}$	Set node and element dump time interval for printing.
PLTT $\Delta time_{plot}$	Set node and element dump time interval for ORION.
SBRF *steps*	Set number of time *steps* between restart dumps.
SIAR *interval*	Set step *interval* for automatic rezoning.
MSRF *reformations*	Set number of stiffness matrix *reformations* per time step.
NSMD *flag*	Set standard solution method *flag*.
DCTOL *tolerance*	Set convergence *tolerance* on displacements.
ECTOL *tolerance*	Set convergence *tolerance* on energy.
NBSR *steps*	Set time *steps* between stiffness matrix reformations.
NBEI *steps*	Set number of time *steps* between equilibrium iterations.
NIBSR *equilibrium_iterations*	Set *iterations* between stiffness matrix reformations.
NAUS *steps*	Set number of arc length unloading *steps*.
IAUM *method*	Set arc length unloading *method*.
IADC *flag*	Set arc length displacement control *flag.*
IADR *direction*	Set *direction* for nodal arc length displacement control.
IACN *method*	Set arc length constraint *method.*
IADM *flag*	Set arc length damping *flag*.
ASIZ *size*	Set initial arc length *size*.
LST *tolerance*	Set line search *tolerance*.
SST *tolerance*	Set slideline stiffness insertion *tolerance*.
RFFC *factor*	Set reduction *factor* for frictional slideline.
RLT *tolerance*	Set rezoner least squares fit *tolerance*.
IGS *flag*	Set geometric stiffness *flag*.

NIKE2D ISLAND Template Commands (p. 62)

DCTOL *tolerance*	Set convergence *tolerance* on displacements.
DELTA *size* $_{time\ initial}$	Set initial time step *size.*
DSTOL *tolerance*	Set step displacement *tolerance*.
DTMAX *size* $_{maximum\ time\ step}$	Set maximum time step *size*.
DTMIN *size* $_{minimum\ time\ step}$	Set minimum time step *size*.
ECTOL *tolerance*	Set convergence *tolerance* on energy.
MAXSTEPS *steps*	Set maximum number of time *steps*.
MAXTRIES *changes*	Set maximum number of time step size *changes*.
MSRF *reformations*	Set maximum stiffness matrix *reformations* per time step.
NGOODSTEPS *steps*	Set number of time *steps* for time step size changes.
NIBSR *equilibrium_iterations*	Set max. *iterations* between stiffness matrix reformations.
NUMREF *reformations*	Set number of *reformations* for a good step.
RCTOL *tolerance*	Set *tolerance* on residuals.
SBRF *steps*	Set number of time *steps* between restart dumps.
TERM *time*	Terminate calculation at *time*.
TSSF *scale*	Set *scale* factor for computed time step size.

TOPAZ2D Control (p. 62)

TITLE	Define title of MAZE output file.
IGM *type*	Set geometry *type*.
BWMO *flag*	Set bandwidth minimization *flag*.
NSMD *method*	Set solution *method*.
CGCTOL *tolerance*	Set conjugate gradient convergence *tolerance*.
RTYPE *type*	Set radiation calculation *type*.
ANALYSIS *type*	Set analysis *type*.
STEP *code*	Set time step *code*.
PRTI *step_interval*	Set node and element dump *step interval* for printing.
PLTI *step_interval*	Set node and element dump *step interval* for ORION.
PRTT $\Delta time_{print}$	Set node and element dump time interval for printing.
PLTT $\Delta time_{plot}$	Set node and element dump time interval for ORION.
SBRF *steps*	Set number of time *steps* between restart dumps.
ALPHA γ	Set Newmark parameter γ:
START $time_{initial}$	Set initial problem time.
TERM $time_{final}$	Set termination problem time.
DELTA *size*	Set time step *size.*
DTMIN $size_{minimum\ time\ step}$	Set minimum time step *size*.
DTMAX $size_{maximum\ time\ step}$	Set maximum time step *size*.
TMPMAX *temperature*	Set maximum *temperature* change in each time step.
TSSF *parameter*	Set time step control *parameter.*
NONL *type*	Set *type* of problem.
MSRF *reformations*	Set maximum coefficient matrix *reformations* per time step.
NIBSR *equilibrium_iterations*	Set *iterations* allowed between coefficient matrix reformations.
TCTOL *tolerance*	Set convergence *tolerance.*
RELAX *parameter*	Set divergence control *parameter.*

CHEMICAL TOPAZ2D Control (p. 66)

TITLE	Define title of MAZE output file.
IGM *type*	Set geometry *type*.
BWMO *flag*	Set bandwidth minimization *flag*.
NSMD *method*	Set solution *method*.
CGCTOL *tolerance*	Set conjugate gradient convergence *tolerance*.
RTYPE *type*	Set radiation calculation *type*.
ANALYSIS *type*	Set analysis *type*.
STEP *code*	Set time step *code*.
IPRTI *step_interval*	Set node and element dump *step interval* for printing.
IPLT *step_interval*	Set node and element dump *step interval* for ORION.
SBRF *steps*	Set number of time *steps* between restart dumps.
ALPHA γ	Set Newmark parameter γ:
RPRT $\Delta timeprint$	Set node and element dump *time interval* for printing.
RPLT $\Delta timeplot$	Set node and element dump *time interval* for ORION.
START *time initial*	Set initial problem *time*.
TERM $time_{final}$	Set termination problem *time*.

DELTA *size* $_{time\ initial}$	Set initial time step *size.*
DTMIN *size* $_{minimum\ time\ step}$	Set minimum time step *size*.
DTMAX *size* $_{maximum\ time\ step}$	Set maximum time step *size*.
TMPMAX *temperature*	Set maximum *temperature* change in each time step.
TSSF *parameter*	Set time step control *parameter.*
NONL *type*	Set *type* of problem.
MSRF *reformations*	Set maximum number of coefficient matrix *reformations*.
NIBSR *equilibrium_iterations*	Set *iterations* allowed between coefficient matrix reformations.
TCTOL *tolerance*	Set convergence *tolerance*.
RELAX *parameter*	Set divergence control *parameter.*

CHEMICAL TOPAZ2D Chemistry Control (p. 68)

NRX *n*	Set *n*umber of chemical reactions.
NMIX *materials*	Set number of mixture *materials*.
MTMD *flag*	Set material property calculation *flag*.
CHMT *materials*	Set number of *materials* used in chemistry.
GPLC *type*	Set *type* of chemistry composition calculation.
CTIN *type*	Set *type* of temporal solution scheme for chemistry.
NRX2 *number*	Set *number* of chemical reactions of type 2.
EUBG *coming_from_material*	Set *"coming from" material* for eff. Eulerian calculation.
EUED *going_to_material*	Set *"going to" material* for eff. Eulerian calculation.
EUAL *tolerance*	Set error *tolerance* for effective Eulerian calculation.
EUAR *area*	Set reaction front *area* using eff. Eulerian calculation.

CHEMICAL TOPAZ2D Reaction Control (p. 69)

GASS *constant*	Set gas *constant.*
ERRX *tolerance*	Set convergence *tolerance* for chemical reaction rates.
ITRX *iterations*	Set maximum number of Newton-Raphson *iterations* to converge chemical reaction rates.
PACT *flag*	Set pressure active *flag*.
MPL1 *species*	Set first *species* number to be plotted.
MPL2 *species*	Set second *species* number to be plotted.
MINT *temperature*	Set minimum *temperature* of reaction.
MAXT *temperature*	Set maximum *temperature* of reaction.
PON *pressure*	Set initial / final *pressure*.
PMAX *pressure*	Set maximum *pressure* of reaction.

CHEMICAL TOPAZ2D Reaction Data (p. 70)

REAC	Initiate reaction process.
CTYPE *type*	Set chemical reaction *type*.
RMNT *temperature*	Set minimum *temperature* of reaction.
RMIT *value*	Set minimum *value* of 1/T of reaction..
RKK *logarithm*	Set *logarithm* of collision frequency.
EEX *energy*	Set activation *energy* of collision frequency.
PFAC *exponent*	Set pressure prefactor *exponent*.
VEX *volume*	Set activation *volume*.
MIXN $n\ p_1 \ldots p_n$	Establish list of reacting materials.
REAC	Initiate reaction process.
STOC $n\ s_1 \ldots s_n$	Establish list of stoichiometric values.
STOS $m\ n\ s_m \ldots s_n$	Establish reaction numbers.
IPRF $n\ c_1 \ldots c_n$	Establish list of composition exponents of reaction.
IPRS $m\ n\ c_m \ldots c_n$	Establish reaction numbers.

Transition From Phase II To Phase III

Analysis Code Establishment (p. 71)

WBCD *format*	Prepare disk file containing mesh data written in accordance with specifications required by analysis code.

Phase III

Graphics Commands (p. 72)

ELPLT	Display element numbers on mesh of materials.
FSON / FSOFF	Enable / disable display of free surfaces and slideline interfaces.
G	Display mesh with all material numbers.
GO	Display mesh to the right of centerline and display outline to the left of centerline.
M *material*	Display material in all plots.
MNON / MNOFF	Enable / disable the display of material numbers.
MO	Display outline of material.
NDPLT	Display node numbers on mesh of material.
O	Display outlines of all materials.
V	Display mesh of material.

General Commands (p. 72)

CMN $e_1\ e_2$ *material*	Change material numbers of elements.
CSF *factor*	Scale nodal coordinates .
CSHF $\Delta r\ \Delta z$	Translate nodal coordinates.
NEOS *node_offset element_offset*	Establish node and element offset.
PHS2	Return to Phase II command section of MAZE.

Initial Nodal Temperatures: NIKE2D - TOPAZ (p. 73)

ERIT *first last step temperature*	Assign initial / reference temperatures of all nodes associated with elements from *first* through *last*.
MRIT *material temperature*	Assign initial / reference temperature of all nodes in material .
NRIT *first last step temperature*	Assign initial / reference temperature of all nodes in the range from *first* through *last* inclusive by step size.
T0 *temperature*	Assign initial / reference temperature of all nodes.
TIC *m n temperature*	Assign initial temperature condition to nodes.
TICV r_1 z_1 ... r_4 z_4 *temperature*	Assign initial temperature boundary condition.

Material Commands (p. 74)

DBQT *type*	Change default value of bulk viscosity type.
DHGQ Q_h	Change default value of hourglass viscosity.
DHQT *type*	Change default value of hourglass stabilization method.
DQL Q_l	Change default value of linear bulk viscosity.
DQQ Q_q	Change default value of quadratic bulk viscosity.
ENDMAT	Terminate current material type definition.
MAT *n type*	Establish material number *n* of material type.
MT *type* (TOPAZ)	Establish material type.
TMAT *n heading* (TOPAZ)	Establish material definition consisting of material number and a heading that is to be placed on the next input line.

Equation-of-State Commands: DYNA2D (p. 75)

ENDEPOS	Terminate equation-of-state specification.
EOS *material_number type*	Define equation-of-state.
HEAD *heading*	Equation-of-state *heading*.

Material Commands - Verbatim Mode (p. 76)

mat material_number 2000 (DYNA2D, NIKE2D)	Material verbatim mode
tmat material_number 2000 (TOPAZ2D)	Material verbatim mode

11 APPENDICES

11.1 Reading Line Segment Data

MAZE can read a file containing a series of points to be connected together to form straight line segments. The optional file, *ifile*, is specified on the MAZE command line: maze c=*cfl* i=*inf*. Each of the following commands will cause MAZE to read the line segment data.

RLN	Read the next line segment definition in the optional input file and use it to create a new line.
RLNS	Read all line segment definitions in the optional input file and use each definition to create a new line.
RSEG	Read the next line segment definition in the optional input file and add its points to the current line.

Each line segment definition must be of the following form. No termination indicator is required for this file. Blank lines or spaces may be used to separate line and segment definitions but these are not required.

The header format is presented in the following table. A header format is required for each group of data..

Header Format

Columns	Definition	Format
1 - 5	Number of points in line segment *nps*: < 0: *nps* set to |*nps*|; order of points reversed as line segment is read in.	I5
6 - 10	Coordinate switch: EQ. 0: r,z EQ. 1: z,r EQ. 2: Θ,*l*	I5
11 - 15	Reflected segment option: EQ. 0: line segment is not reflected EQ. 1: segment is reflected about line segment parallel to *r*-axis EQ. 2: generate a single continuous line segment with both the nonreflected and reflected segments	I5

Columns	Definition	Format
16 - 25	Scale factor for coordinate values. Default: 1	E10.0
26 - 35	*z*-coordinate shift [Shift applied before line segment reflected.]	E10.0
36 - 45	*z*-value of axis about which line segment is reflected	E10.0
46 - 55	*r*-coordinate shift	E10.0

.

Data Format

Columns	Definition	Format
1 - 10	*r*-coordinate [coordinate switch EQ 0] *z*-coordinate [coordinate switch EQ 1] Θ [coordinate switch EQ 2]	E10.0
11 - 20	*z*-coordinate [coordinate switch EQ 0] *r*-coordinate [coordinate switch EQ 1] *l* [coordinate switch EQ 2]	E10.0

When the Θ,*l* description is used, the *r*-coordinate is specified as *l*cos(Θ - 90) and the *z*-coordinate is specified as *l*sin(Θ - 90). Angle Θ is specified in degrees and is measured counterclockwise from the negative *z*-axis.

11.2 Importing Finite Element Geometry Data

MAZE can read an ASCII file containing finite element geometry data and convert this data into line and part definitions. Two file formats are available: the default DYNA2D, NIKE2D, and TOPAZ2D format and a "neutral" format. Multiple parts can be stored in each geometry file. Imported parts are differentiated by their material number. Geometry files can be imported by using a start-up command line argument: maze g=*gfl* or with the interactive command **GEOM**.

GEOM *gfl*	Import finite element geometry data from file *gfl*. File *gfl* must be specified on the line following command GEOM.

Command GEOM will re-number the imported nodes and elements. Imported node numbers do not need to be entered sequentially nor must all node numbers be used by the element topologies.

GEOM File Format:
Card #1

Columns	Definition	Format
1 - 5	Number of nodes	I5
6 - 10	Number of elements	I5
11 - 15	File format flag: EQ. 0: DYNA2D, NIKE2D, or TOPAZ2D format (default) EQ. 1: neutral format	I5

GEOM File Format:
Cards #2 - number of nodes + 1

Format Flag	Columns	Definition	Format
0	1 - 5	Node number	I5
0	11 - 20	r-coordinate	e10.0
0	21 - 30	z-coordinate	e10.0
1	1 - 5	Node number	I5
1	11 - 20	r-coordinate	e15.0
1	21 - 30	z-coordinate	e15.0

GEOM File Format:
Cards #(number of nodes + 2) to (number of nodes + number of elements + 1)

Format Flag	Columns	Definition	Format
0	6 - 10	Node number 1	I5
0	11 - 15	Node number 2	I5
0	16 - 20	Node number 3	I5
0	21 - 25	Node number 4	I5
0	26 - 30	Material number	I5
1	1 - 5	Node number 1	I5
1	6 - 10	Node number 2	I5
1	11 - 15	Node number 3	I5
1	16 - 20	Node number 4	I5
1	21 - 25	Material number	I5

12 ACKNOWLEDGEMENTS

The author wishes to acknowledge the many suggestions and requests from the MAZE user community. Collaborative feedback from analysts and researchers, both inside Lawrence Livermore National Laboratory and in industry, has been invaluable in contributing to this manual and to the continuing evolution of MAZE.

In particular, thanks to Peter Raboin, Edward Zywicz, and Thomas Healy for their many suggestions and contributions, particularly in the structure and organization of the manual, theoretical background of the material models in DYNA2D and NIKE2D, and example problems. The assistance of Albert Nichols, Jeffrey Hendricks, and Arthur Shapiro for providing theoretical background on TOPAZ2D and CHEMICAL TOPAZ2D and examples is gratefully acknowledged.

The contributions of John Hallquist, J. Levatin, R. Rosinsky, Robert Bailey, and A. Harral for their efforts in developing the previous editions of the MAZE User Manual are gratefully acknowledged. The contribution of H. Nebeck in providing geometric definitions and illustrations is also acknowledged. Much of the command definition structure and illustration formats presented in this manual are based upon the efforts and work of these individuals.

The author also wish to acknowledge his colleagues at Lawrence Livermore National Laboratory. Gerald Goudreau, Robert Whirley, Bruce Engelmann, Douglas Faux, Larry McMichael, Filbert Diaz, and Douglas Speck for their ideas, technical insight, and encouragement. In particular, the author wishes to once again acknowledge Peter Raboin for patiently reviewing the manual in all phases of development and providing many constructive suggestions leading to a more useful manual.

Finally, the author wishes to again acknowledge John Hallquist for his pioneering work in the development of the MAZE input generator for the LLNL analysis codes.

13 REFERENCES

Christon, M. A., and Dovey, D., INGRID: A 3-D Mesh Generator for Modeling Nonlinear Systems - User Manual. University of California, Lawrence Livermore National Laboratory, UCRL-MA-109790, September, 1992.

Dovey, D., and Spelce, T. E., GRIZ: Finite Element Analysis Results Visualization for Unstructured Grids - User Manual. University of California, Lawrence Livermore National Laboratory, UCRL-MA-115696, October, 1993.

Engelmann, B. E., and Hallquist, J. O., NIKE2D: A Nonlinear, Implicit, Two-Dimensional Finite Element Code for Solid Mechanics - User Manual. University of California, Lawrence Livermore National Laboratory, UCRL-MA-105413, April, 1991.

Engelmann, B. E., and Whirley, R. G., ISLAND: Interactive Solution Language for an Adaptive NIKE Driver - User Manual. University of California, Lawrence Livermore National Laboratory, UCRL-MA-108721, October, 1991.

Engelmann, B. E., Whirley, R. G., and Shapiro, A.B., PALM2D: A Nonlinear Finite Element Program for the Coupled Thermomechanical Response of Solids in Two Dimensions. University of California, Lawrence Livermore National Laboratory, UCID-21868, March, 1990.

Hallquist, J. O., MAZE: An Input Generator for DYNA2D and NIKE2D. University of California, Lawrence Livermore National Laboratory, UCID-19029, Rev. 2, June, 1983.

Hallquist, J. O., and Levatin, J. L., ORION: An Interactive Color Post-Processor for Two Dimensional Finite Element Codes. University of California, Lawrence Livermore National Laboratory, UCID-19310 Rev. 2, August, 1985.

Maker, B. N., NIKE3D: A Nonlinear, Implicit, Three-Dimensional Finite Element Code for Solid and Structural Mechanics - User's Manual. University of California, Lawrence Livermore National Laboratory, UCRL-MA-105268 Rev. 1, April, 1995.

Nebeck, H. E., Geometry Definition with MAZE. University of California, Lawrence Livermore National Laboratory, UCID-20850, August, 1986.

Nichols, A. L., and Westerberg, K. W., Chemical TOPAZ: An Addenda to the TOPAZ Manual - UCRL-ID 104558 Modifications to the Heat Transfer Code TOPAZ: The Addition of Chemical Reaction Kinetics and Chemical Mixtures. University of California, Lawrence Livermore National Laboratory, UCRL-ID-104558 Add 1, January, 1994.

Shapiro, A. B., TOPAZ3D - A Three-Dimensional Finit Element Heat Transfer Code. University of California, Lawrence Livermore National Laboratory, UCID-20484, August, 1985.

Shapiro, A. B., and Edwards, A. L., TOPAZ2D Heat Transfer Code Users Manual and Thermal Property Data Base. University of California, Lawrence Livermore National Laboratory, UCRL-ID-104558, April, 1992.

Spelce, T. E., TAURUS: An Interactive Post-Processor for the Analysis Codes NIKE3D, DYNA3D, and TOPAZ3D. University of California, Lawrence Livermore National Laboratory, UCRL-MA-105401, May, 1991.

Whirley, R. G., and Engelmann, B. E., DYNA2D: A Nonlinear, Explicit, Two-Dimensional Finite Element Code for Solid Mechanics - User Manual. University of California, Lawrence Livermore National Laboratory, UCRL-MA-110630, April, 1992.

Whirley, R. G., and Engelmann, B. E., DYNA3D: A Nonlinear, Explicit, Three-Dimensional Finite Element Code for Solid and Structural Mechanics - User Manual. University of California, Lawrence Livermore National Laboratory, UCRL-MA-107254 Rev. 1, November, 1993.

INDEX